Learning Common Features from fMRI Data of Multiple Subjects

John Ramish
Advised by Prof. Tom Mitchell
8/10/04

Abstract

Functional Magnetic Resonance Imaging (fMRI), a brain imaging technique, has allowed psychologists to identify what parts of the brain are involved in various tasks. Recently, Mitchell et al (2003) have used fMRI in a novel way: to infer from it a person's mental states using machine learning algorithms. Wang, Hutchinson, and Mitchell (2003) have extended these algorithms to make predictions across subjects, using hard-coded common representations. We have gone further to develop cross-subject clustering, a method of learning common representations. This method not only offers the theoretical advantage of learning, but also appears to offer the empirical advantage of improved cross-subject predictions. The empirical studies were limited to a single dataset (Sentence-then-Picture), however, so further work is needed to confirm its general utility. Several of our other experiments demonstrate that unsupervised learning generally attains accuracies nearly as high as those of supervised learning across subjects, and in some cases higher. Finally, we briefly catalog several other less successful approaches to the cross-subject prediction problem.

Outline

1 Introduction

Functional Magnetic Resonance Imaging (fMRI), a brain imaging technique, has allowed psychologists to identify what parts of the brain are involved in various tasks. Recently, Mitchell et al (2003) have used fMRI in a novel way: to infer from it a person's mental states using machine learning algorithms. This research could eventually lead to a "virtual sensor" of mental state, which would grant unprecedented insight into the human mind.

Wang, Hutchinson, and Mitchell (2003) have extended this research to make predictions across subjects. They studied three representations common across subjects:

1) *ROI average:* the average activation of all voxels in each Region of Interest (ROI)
2) *ROI active average (n):* the average activation of the n most active voxels in each ROI
3) *Talairach coordinates:* the standard brain coordinate system

With these common representations, classifiers trained on data from several training subjects tested well on data from new subjects.

Yet despite this good performance, these common representations have several limitations. Most notably they hard-coded. Hence they lack the flexibility of learned common representations. Furthermore, they require the knowledge of experts in neuroanatomy to map out corresponding spatial regions on different brains. And arguably, common representations shouldn't be based on spatial regions at all. Instead, they should feature the common types of activation behaviors, wherever they may be found.

This project addressed these shortcomings of the earlier approaches with the method of cross-subject clustering, described in the next section. We also discuss several less successful alternative approaches in section 3.

2 Cross-subject clustering: learning common features

2.1 Algorithm Description

This algorithm learns a representation common across subjects by partitioning the collective voxels of all subjects into clusters, which have associated Gaussian distributions for each class at each time step of the experiment. Simultaneously, it learns the maximum likelihood class labels of any unlabeled examples from any subjects.

To initialize, the algorithm randomly assigns class labels to unlabeled examples, initializes each cluster with a randomly selected single voxel, and assigns each voxel to the cluster that maximizes the likelihood of its data. Then it iterates, EM-style:

1) Given all example class labels and voxel cluster assignments, for each cluster find the Gaussian distributions that maximize the likelihood of all of its voxels' data.
2) Given all clusters' distributions and example class labels, for each voxel for each subject, find the cluster that maximizes the likelihood of its data.
3) Given all clusters' distributions and voxel cluster assignments, for each unlabeled example of each subject, find the class label that maximizes the likelihood of all of the voxels' data.

Pictures of the distributions for five clusters before and after learning are shown in figures 1 and 2. A less successful alternative approach clustered voxels based on the Euclidean distances between their time courses and used the means of their time courses as cluster centers (see section 3.3). The idea of using a distance metric that incorporated the example class labels was inspired by the approach of Toft et al (1997).

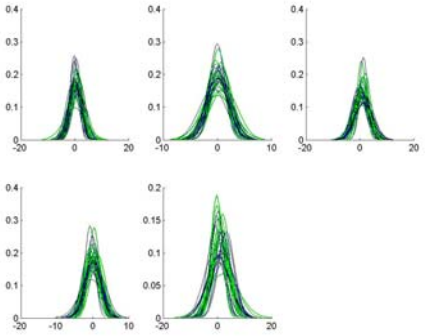

Figure 1. Five clusters after initialization. For each cluster, the Gaussian distributions from all 16 time courses are shown superimposed.

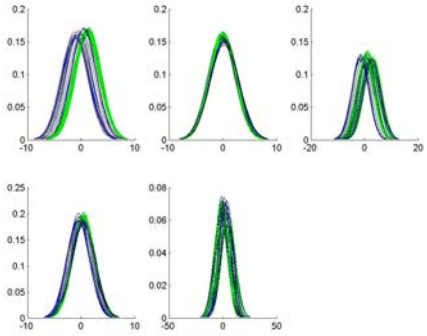

Figure 2. The same five clusters after ten iterations. The two classes (blue and green) are better separated within the clusters.

2.2 Results

For an empirical comparison of these algorithms, we assessed three types of classifiers: single-subject classifiers, leave-one-subject-out classifiers, and unsupervised classifiers. Single-subject classifiers (using leave-one-example-out accuracies) provide a baseline accuracy for evaluating cross-subject predictors. Leave-one-subject-out classifiers train on all but one subject and test on that subject. Unsupervised classifiers, both single-subject and cross-subject, provide an alternative approach to making predictions about new subjects. We experimented with all of these methods using the Sentence-then-Picture data and variably on data from all twelve subjects and from only the five best subjects. All classifiers used voxels normalized by the "transformIDM_normalizeTrials" function. Below are brief descriptions of all of the classifiers and tables of the accuracies on all twelve subjects and on the best five.

Single-subject classifiers

Single-subject ROI active 20 average[*] *(SS ROIact):* Gaussian Naïve Bayes (GNB) applied to the averages of the 20 most active voxels in each ROI

Single-subject active 200 [*] *(SS act):* GNB applied to the 200 most active voxels

Single-subject clustering active 200 (SS cl): clustering (as in cross-subject clustering, but with voxels from a single-subject) applied to the 200 most active voxels using 5 clusters and 5 iterations

Single-subject ROI average[*] *(SS ROIall):* GNB applied to the averages of all voxels in each ROI

Leave-one-subject-out classifiers

L1SO cross-subject clustering active 200 (L1SO cl): cross-subject clustering applied to the 200 most active voxels, using 15 clusters and 10 iterations

[*] These classifiers were previously studied by Wang, Hutchinson, and Mitchell (2003), and were reproduced for these experiments. We could not exactly reproduce the earlier results: our accuracies are significantly lower. The most likely source of the discrepancy is in the subtleties of normalization. Other possible sources of differences are our use of 12 subjects instead of 13 and the subtleties of the selection of active voxels.

L1SO ROI average[*] *(L1SO ROIall):* GNB applied to the averages of all voxels in each ROI

L1SO ROI active 20 average[*] *(L1SO ROIact):* GNB applied to the averages of the 20 most active voxels in each ROI

Unsupervised classifiers

Unsupervised cross-subject clustering active 200 (UCS cl): cross-subject clustering applied to the 200 most active voxels with no class labels given for any examples for any subjects. The resulting representation is common across subjects. The algorithm used 15 clusters and 10 iterations.

Unsupervised single-subject EM spherical ROI active 20 average (USS ROIact): EM for Gaussian Mixture Models applied to the averages of the 20 most active voxels in each ROI for each subject individually. EM learned one multivariate Gaussian for each class. The covariance matrices of the Gaussians were assumed spherical. This is the unsupervised version of GNB with the distributions for all voxels at all time steps for both classes assumed to have equal variances. We used the Netlab toolbox for Matlab for the EM algorithm (available at http://www.ncrg.aston.ac.uk/netlab/).

Unsupervised single-subject clustering active 200 (USS cl): clustering applied to the 200 most active voxels for each subject individually (to yield subject-specific representations). The algorithm used 5 clusters and 10 iterations.

All 12 subjects

	Subject												**Mean**
	1	2	3	4	5	6	7	8	9	10	11	12	
SS ROIact	0.9750	1.0000	1.0000	1.0000	0.9000	0.8000	0.8500	0.9000	0.8750	1.0000	0.8500	1.0000	**0.9292**
SS act	0.9000	1.0000	0.9750	1.0000	0.9000	0.8750	0.8250	0.9500	0.9000	1.0000	0.8000	1.0000	**0.9271**
SS cl	0.8500	1.0000	0.9500	0.9750	0.9000	0.7500	0.8500	0.9000	0.9000	1.0000	0.7750	1.0000	**0.9042**
SS ROIall	0.8750	0.9000	0.9750	1.0000	0.9250	0.6750	0.8750	0.8000	0.9250	0.9750	0.7750	1.0000	**0.8917**
L1SO cl	0.7500	0.9750	1.0000	1.0000	0.9250	0.8500	0.8750	0.9000	0.9000	1.0000	0.8000	1.0000	**0.9146**
L1SO Rall	0.6250	0.7750	0.9250	1.0000	0.9250	0.7250	0.8500	0.7750	0.8250	0.9750	0.8000	0.9250	**0.8438**
L1SO Ract	0.5750	0.8750	0.9500	1.0000	0.9000	0.8000	0.8250	0.7250	0.8000	0.9500	0.8500	0.8750	**0.8437**
UCS cl	0.6500	0.9250	0.9750	1.0000	0.9250	0.8500	0.8750	0.9250	0.8750	0.9750	0.5250	1.0000	**0.8750**
USS Ract	0.6250	1.0000	1.0000	1.0000	0.9000	0.8000	0.6000	0.9250	0.8750	0.9750	0.7750	1.0000	**0.8729**
USS cl	0.6250	0.9500	0.9250	1.0000	0.9000	0.6250	0.8250	0.6250	0.8750	1.0000	0.7750	1.0000	**0.8438**

Best 5 subjects (3, 4, 9, 10, 12)

	Subject					**Mean**
	3	4	9	10	12	
SS ROIact	1.0000	1.0000	0.8750	1.0000	1.0000	**0.9750**
SS act	0.9750	1.0000	0.9000	1.0000	1.0000	**0.9750**
SS cl	0.9500	0.9750	0.9000	1.0000	1.0000	**0.9650**
SS ROIall	0.9750	1.0000	0.9250	0.9750	1.0000	**0.9750**
L1SO cl	0.9750	1.0000	0.9000	1.0000	1.0000	**0.9750**
L1SO ROIall	0.9750	1.0000	0.9000	0.9750	1.0000	**0.9700**
L1SO ROIact	1.0000	1.0000	0.8750	0.9750	0.9750	**0.9650**
UCS cl	1.0000	1.0000	0.8750	1.0000	1.0000	**0.9750**
USS ROIact	1.0000	1.0000	0.8750	0.9750	1.0000	**0.9700**
USS cl	0.9250	1.0000	0.8750	1.0000	1.0000	**0.9600**

2.3 Analysis

Several trends are apparent in the data:

1) Excellent performance of cross-subject clustering

The L1SO cross-subject clustering algorithm is easily the best predictor for new subjects for the data from all 12 subjects. It achieves 91.5% accuracy, about 7% better than the L1SO ROI classifiers (84.4 %) and 4% better than the unsupervised classifiers (87.5%). Moreover, it performs better than the corresponding single-subject clustering classifier's accuracy of 90.4% and nearly as well as the original single-subject GNB classifier (SS act), which had 92.7% accuracy.

The L1SO cross-subject clustering algorithm performs similarly well on the 5 best subjects, making it the best predictor for new subjects on this dataset.

2) Excellent performance of unsupervised learners

All three unsupervised classifiers perform at least as well as the L1SO supervised classifiers based on the ROI representations. Two of them achieve accuracies of 87.5%, about 3% higher than the accuracies of the L1SO ROI classifiers (84.4%), and only 4% lower than the accuracy of the best cross-subject predictor (L1SO cross-subject clustering, with 91.5%). In the case of the ROI representation, it's interesting to observe that it's better to use unsupervised learning to look for patterns within a subject's own data than to try to make predictions based on known patterns from other subjects. Cross-subject clustering, however, benefits from labeled data from other subjects. In fact, it even benefits from unlabeled data from other subjects: for unsupervised learning, using cross-subject clusters outperforms using subject-specific clusters.

3) Excellent performance of all methods on 5 best subjects

At least on this dataset, for the best 5 subjects the choice of cross-subject classifier is inconsequential: all achieve about 97% accuracy, which is as good as the single-subject predictor accuracies.

4) ROI methods suffer from negative transfer, but cross-subject clustering does not

As noted above, when restricted to the five best subjects, all methods perform similarly well at about the 97% accuracy level. Yet the ROI methods perform significantly worse on these subjects when the other 7 subjects are included: accuracies drop from 97% to 93% and from 96.5% to 91.5%. In effect, there is negative transfer from the bad subjects. The cross-subject clustering approaches, however, resist this effect: one accuracy drops slightly from 97.5% to 96.5% and the other actually increases from 97.5% to 98%!

2.4 Cross-subject clustering vs. ROI approaches

Advantages of cross-subject clustering

1) Better classification accuracies

On all 12 subjects for the Sentence-then-Picture data, the L1SO cross-subject classifier achieved an accuracy of 91.5%, about 7% better than the ROI approaches, which both had accuracies of 84.4%.

2) Learned common representation

Cross-subject clustering, unlike the ROI approaches, learns its common representation. This provides autonomy and adaptability: no expert is needed to identify ROIs or Talairach coordinates, and the common representation is specifically suited to the given classification task.

3) Activation-based common representation

Arguably, it's better to extract common features by similar behavior than by similar location. Cross-subject clustering does this. ROI representations and Talairach coordinates do not.

4) Representation of individual differences and selective transfer

Cross-subject clustering allows any distribution of voxels among clusters. As a result, if one subject has more voxels of a particular type, it can represent that individual difference. It can also naturally incorporate similarities and differences among subjects through the similarities and differences of their voxel distributions. Selective transfer results.

The data evidence this effect. As noted in the previous subsection, unlike ROI methods, cross-subject clustering resists negative transfer.

Finally, we note that the balance between subject conformity and individual freedom in the common representation can be regulated by the number of clusters. Fewer clusters enforce greater conformity.

5) Variable resolution

By changing the number of clusters, one can control the "resolution" of cross-subject clustering. By contrast, ROI representations have fixed resolution, given by the number of ROIs.

Disadvantages of cross-subject clustering

1) Classifier-specific common representation

The cross-subject clustering algorithm presented here is based upon the Gaussian naïve maximum likelihood classifier: the clusters are chosen so as to maximize the likelihood of the data under the naïve Gaussian model. By contrast, the ROI and Talairach coordinate representations can easily be combined with any classifier.

This limitation can be addressed by several extensions of the algorithm, which are discussed in the next section.

2) Limited form of common representation

The cross-subject clustering common representation is based on groups of voxels. Granted, the ROI representations are similarly limited to averages of groups of voxels. But there are many other conceivable common representations. We note that the ROI representations could easily be adapted to be based on groups of voxels (sharing Gaussian distributions among voxels) instead of averages of groups of voxels. Similarly,

the cross-subject clustering algorithm can be adapted for other types of common representations. See the next section.

3) Slowness

The cross-subject clustering algorithm is significantly slower than the ROI approaches. A single run for all 12 subjects using 15 clusters and 10 iterations can take 10 minutes. Hence the time for computing all leave-one-subject-out accuracies can be a few hours. By contrast, the ROI approaches can do this in less than a minute. Of course, this might be expected since the cross-subject clustering algorithm has to learn its common representation, while the ROI approaches have hard-coded common representations.

2.5 Extensions and variations

The cross-subject clustering algorithm is composed of three parts: the algorithm for searching the space of possible clusterings, the type of feature derived from each cluster, and the performance measure of the resulting features. It uses an EM-like procedure to search for clusters, groups voxels in each cluster, and measures the effectiveness of a grouping by the likelihood of the data under the maximum-likelihood naïve Gaussian model. Although there are not any obvious better search algorithm alternatives, there are several promising alternative features and measures of features:

Alternative features from clusters

One alternative feature type is the average of the group of voxels in a cluster. Some preliminary experiments suggested this did not do as well as simply grouping.

Alternative measures of features

Most measures attempt to quantify the separation of the data in the resulting feature space. Several statistical measures, based on distances of means and inter- and intra- class variances, can be used. In some sense, these are independent of the classifier, and the resulting features could be used by any classifier. Another way to quantify the resulting separation of the data is by the training success of a classifier. This algorithm uses the likelihood of the data under the maximum likelihood naïve Gaussian model. One could easily substitute the margin of a SVM classifier, or a measure based on the k-nearest neighbor classifier.

There are also a number of other simple extensions of the algorithm:

Select n most useful voxels

There is no reason why every voxel must belong to a cluster. It would be easy to select only the n most useful voxels. On each iteration, after assigning all voxels to clusters, the n most useful voxels could be identified by the likelihood of their data under the current model. The rest could be discarded for the rest of the iteration (hence having no effect on the new example class labels or new clusters' distributions).

Incorporate spatial bias: cluster by ROI

To incorporate a spatial bias, one can perform the clustering separately for each ROI. Preliminary experiments on this method suggested it did not work as well.

Online version

This version of the algorithm is offline: it clusters the test subject's voxels with those of the training subjects. An alternative is the online approach, in which the training subjects are clustered first and then the test subject is added.

3 Other approaches

3.1 Colearning neural networks

We first explored this multi-subject learning model in a Spring 2004 Independent Study project. The model, shown in figure 3, has individual linear transformations to a common representation for each subject and a shared function mapping these common features to the cognitive state classification.

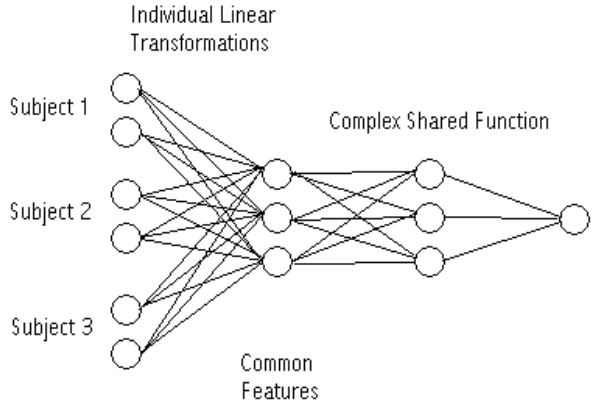

Figure 3. The colearning neural network model.

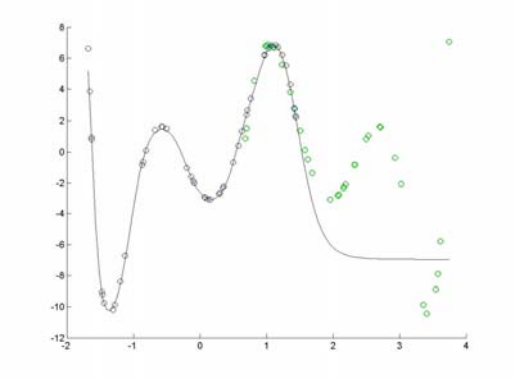

Figure 4. Here the curve has been fit to the data of one subject (the blue points). Gradient descent was unable to fit the data of another subject (the green points) to the learned model. Instead, it found a local minimum that matches only the largest bumps.

The model can represent a large space of features: all linear transformations of the voxel inputs. In particular, this space includes both the ROI average and ROI active average representations.

The Spring 2004 project suggested the model needed bias to cope with only 40 examples per subject for an input space of over 100 dimensions. It also revealed the need for a better theoretical understanding of the model.

In response to this need, this project investigated the general properties of the colearning neural net model through experiments on several synthetic datasets. We considered three levels of complexity of the shared function: lines, sine waves, and polynomials. We also gave new subjects both offline and online treatment. In general, the desired common representation was the global minimum, but there were also scads of inferior local minima. Figure 4 shows an example. This difficulty was serious enough on these toy problems to warrant abandoning the model.

Yet in many ways, cross-subject clustering provides what colearning neural nets failed to deliver. By incorporating bias through narrower classes of features and classifiers, it succeeded in learning common features.

3.2 Voxel matching

This algorithm matches individual voxels of different subjects by activation behavior, much the same way that Talairach coordinates matches voxels by spatial location. We experimented with mapping the test subject's voxels to those of the training subject and vice versa. The former worked better. Using this method with several different training subjects produced several alternative classifiers for the test subject. Taking votes from these on class label predictions yielded an effective cross-subject classifier. Here are its accuracies for predictions about the five best subjects using the 200 most active voxels for each, un-normalized:

	Subjects					**Mean**
	1	2	3	4	5	
Voter matching	0.9250	1.0000	0.8750	1.0000	0.9875	**0.9575**

Although this method does well, cross-subject clustering supersedes it. Cross-subject clustering provides a similar activation-based common representation, but with more subjects, with variable resolution, and with better results.

3.3 k-means clustering by Euclidean distance

See Dimitriadou et al (2003) for a comparison of clustering methods for fMRI. Before investigating cross-subject clustering, we experimented with clustering voxels based on the Euclidean distance between their time courses. For single-subjects, this performed better than using all voxels but not as well as using ROI averages. Clearly, the distance metric used for cross-subject clustering is superior.

3.4 ICA, PCA, discriminant analysis

See Calhoun et al (2003) for an overview of the application of ICA to fMRI. Preliminary results of classifiers based on these methods were discouraging. Given the unnormalized data of one subject, FastICA first extracted 200 principal components from the original voxels and then extracted 200 independent components from these. GNB applied to these features achieved an accuracy of 65%, compared to 70% when applied to the original voxels. A preliminary experiment on a form of PCA was similarly disappointing. Based on the limitation that ICA and PCA don't consider class labels, we developed a method to extract components that separate the data by class label. As we later discovered, it was similar to the existing discriminant analysis method. At any rate, it had mixed results.

A fundamental difficulty with these approaches to feature extraction is how to get features common across subjects from them. Perhaps the features for different subjects could be matched.

4 Further research and conclusions

4.1 Further work

This project presents many opportunities for further research. Here are a few possible future research directions:

Experiments on other datasets

This work only evaluated the performance of cross-subject clustering on the Sentence-then-Picture dataset. To confirm its general utility, it needs to be applied to a variety of other datasets.

Extensions of the algorithm

There are many simple and potentially useful extensions of the cross-subject clustering algorithm. See section 2.5 for a description of these.

Combining multiple experiments

To analyze data from multiple experiments using cross-subject clustering, there are two obvious approaches:

1) Cluster across experiments
   This simple extension of the algorithm should identify voxels with similar behavior on a variety of tasks. With many tasks, this should produce a good general-purpose common representation.
2) Cluster for each experiment individually
   Then one can compare the roles of voxels based on cluster memberships for different experiments.

4.2 Conclusions

Based on experiments with the Sentence-then-Picture data, the cross-subject clustering algorithm appears to have both empirical and theoretical advantages over ROI approaches. Further experiments on other datasets are needed to confirm its general utility. Several other experiments demonstrate that unsupervised learning is an effective method of making predictions about new subjects. This shows promise for the goal of automatically identifying mental states.

4.3 Further information

More information on all aspects of this work is easily available. We have a comprehensive log detailing the experiments discussed in this paper and referencing Matlab code and workspaces.

Acknowledgements

We would like to acknowledge the guidance of Prof. Tom Mitchell on this project. He listened patiently, provided valuable feedback, and offered insightful suggestions. This project was funded in part by a Summer Undergraduate Research Fellowship from Carnegie Mellon University.

References

Calhoun, V., Adali, T., Hansen, L. K., Larsen, J., and Pekar, J. (2003). ICA of functional MRI data: an overview. In *Fourth International Symposium on Independent Component Analysis and Blind Source Separation*, pages 281-288, Nara, Japan. Available online at: http://www.kecl.ntt.co.jp/icl/signal/ica2003/cdrom/data/0219.pdf.

Dimitriadou, E., Barth, M., Windischberger, C., Hornik, K., and Moser, E. (2003). A Quantitative Comparison of functional MRI Cluster Analysis. Available online at: www.ci.tuwien.ac.at/~dimi/oenb-papers/AIM2003.pdf

Mitchell, T., Hutchinson, R., Just, M., Niculescu, R.S., Pereira, F., and Wang, X. (2003). Classifying Instantaneous Cognitive States from fMRI Data. *American Medical Informatics Association Symposium*, 2003.

Toft, P., Hansen, L. K., Nielsen, F. A., Goutte, C., Strother, S., Lange, N., Mørch, N., Svarer, C., Paulson, O. B., Savoy, R., Rosen, B., Rostrup, E., and Born, P. (1997). On clustering of fMRI time series. In Friberg et al. (1997), page S456.

Wang, X., Hutchinson, R., and Mitchell, T. (2003). Training fMRI Classifiers to Detect Cognitive States across Multiple Human Subjects. *Neural Information Processing Systems 2003*.