

Pittsburgh Brain Activity Interpretation Competition 2006

Methods Description

A whole-brain correlation-based approach to decoding high-level features in fMRI data.

Abstract

To discover and decode multiple high-level features of fMRI activity of subjects watching feature-salient television clips, we employed a whole-brain correlation-based method for determining feature-related regions of interest, combined with independent component analysis. To train on and predict this data, we used three main algorithms: (a) standard linear regression, (b) sparse logistic regression (a novel Bayesian method by our group), and (c) a hybrid linear classifier-regression algorithm (a novel tree-based method by our group). Using a leave-one-run-out cross-validation test, we achieved an average correlation score (of the first 14 features) of 0.392 when training on movie 1 and testing on movie 2, and a score of 0.308 when training on movie 2 and testing on 1. This demonstrates that an anatomically-blind, correlation-based method of ROI selection combined with ICA and simple linear prediction methods can provide surprisingly powerful tools for identifying feature-related ROI's and predicting high-level features in complex fMRI activity.

Introduction

In previous fMRI studies, we usually performed separate simple task-related runs to localize relevant features to specific brain areas for determining our primary regions of interest (ROI's). Since such runs were not available for this study, and due to the high-level nature of the features, which may be distributed across many areas, we decided to try a new whole-brain correlation-based approach to determine our regions of interest. Because this still generates a very large number of significantly correlated voxels, we employed additional methods of reducing the number of inputs to our prediction algorithms, including independent component analysis (ICA) with PCA reduction and our own sparse logistic regression algorithm (Sato, 2004; Tipping, 2001), after removing small clusters with SPM. Because some features may be more readily localized to a small number of voxels than others, we decided to vary some of these parameters by feature and took the general approach of looping through a large number of parameter combinations for each feature, calculating the best set based on the average correlation of each prediction in a leave-one-run-out cross-validation test. After determining our ROI's, we primarily used standard linear regression for training and prediction, not only because linear methods allow for more straightforward analysis but also because we had discovered (from previous experience) that determining the ROI's appears to play a more important role in prediction than the actual classifier itself. Towards this end, we hope to show that anatomically-blind, correlation-based methods of ROI selection combined with ICA and simple linear prediction methods provide surprisingly powerful tools for identifying areas related to various high-level features.

Methods

We largely focused on the "preprocessed", analyze-format (native-space) data because standard methods of motion correction, slice-time correction, and linear trend removal had been performed. We did not investigate the normalized nor flattened images because such data, while extremely useful for display and traditional analysis, distorts the data at the voxel level.

Because a small smoothing kernel can improve the SN ratio of locally correlated voxels, we did optionally smooth the data with SPM. However, such smoothing did not increase our validation performance (described below), but rather slightly decreased it.

For our analysis, the majority of our work was performed in Matlab using a combination of various SPM 2 and 5 functions, standard Matlab 7 functions, and our own Matlab neural decoding libraries, which we developed for both this and other projects.

Our major processing steps are as follows:

1. Read "preprocessed", analyze-format data for each subject into Matlab using SPM/Analyze-compatible functions.
2. Calculate the mean of each voxel across all runs and apply a threshold to eliminate non-brain areas. This threshold was chosen to be at the trough between non-brain and brain-active peak areas as observed in its histogram (around 300).
3. Read feature ("label") and movie timing data into Matlab, shift movie onset by 2 volumes, and offset times by 5 to align the "blank" interval (which was not convolved) with the homodynamic response. The movie onset shift was less than the offset to avoid removing any movie response, in the feature or activity, shifted by the combined subject response times hdf response delay.
4. Apply a moving (temporal) average of 1 to 3 volumes to the data and shift the set by -1 to 0 volumes to account for shift of the mean. Of the 1-3 window widths and shifts examined, a 3 volume window with a -1 shift appeared to provide the best results (as defined below).
5. Split both feature and fMRI run data into "movie" and "blank" sets (concatenating sections) based on these shifted movie times.
6. To calculate our functional ROI, calculate the correlations of every voxel with every feature. This was done with both the blanks included and excluded. Blank excluded appeared to provide the best results. This resulted in a separate ROI for each feature for each run and a rank of each voxel based on its correlation coefficient. For run 3, the ROI was calculated from the data of runs 1 and 2 concatenated. When superimposed on the whole brain image, plots of some of these ROI's, such as Motion and Language, appeared to correspond the anatomical ROI's (see Figures 1 and 2 in the Appendix).
7. Remove voxels with few neighbors from each ROI with `spm_clusters` using a cluster threshold of 6 or 10. A threshold of 6 provided the best overall results.
8. Combine ROI's into a joint ROI, eliminating duplicates, and filter the fMRI data with the joint ROI to make it more manageable.

9. For each label and each parameter set, take the following steps:
 - a. Select the fMRI data within each ROI based on the training set ROI.
 - b. Reduce this data to the top N correlated voxels based on the rank calculated with the ROI.
 - c. Normalize (z-norm) the activity of each voxel.
 - d. Apply ICA with the fast ICA algorithm (Hyvärinen & Oja, 1997, 1999) using PCA reduction to reduce the vector length from N to P, where P is the number of ICA components to be calculated.
 - e. Train using either run 1 or 2 for the validation test (testing on the other) or on both runs 1 and 2 for predicting run 3.
 - f. Testing using the ICA and regression/classifier weights.
 - g. Calculate the correlations between each prediction and correct feature for each parameter for each feature.

Three main algorithms (and many parameter combinations of each) were used to train and test on the data: (a) standard linear regression, (b) sparse logistic regression (a novel Bayesian method by our group), and (c) novel hybrid linear classifier-regression algorithm.

Optimal parameter combinations of the steps above were found by looping through the above algorithms and some preprocessing steps, calculating their correlation values, and storing those sets with the highest average cross-validation correlation values.

Results and Discussion

Using a leave-one-run-out cross-validation test, we achieved an average correlation score of 0.392 (across the first 14 features as measured by the scoring program) when training on movie 1 and testing on movie 2, and a score of 0.308 when training on movie 2 and testing on 1. When we optimized the parameters on a single train-test set, we could achieve a score of 0.461 for training on run 1 and testing on 2 (Tr1-Te2), and 0.369 for Tr2-Te1. However, to best predict the features of a truly unknown run 3, we chose the set of parameters that provided the best average across our validation set. In general, the addition of optional actor and location features rarely increased our score above the score of the 14 primary features.

Of the various types of preprocessing and regression methods performed, the following set of parameters provided the best overall scores on the validation test across all subjects: “preprocessed”, native, non-smoothed fMRI data; an average activity threshold of 300; blank regions removed before both ROI formation and training; using the top 1000 correlated voxels for each feature; using `spm_cluster` with a threshold of 6; varying the numbers N and P of the inputs to ICA with PCA reduction for each feature and picking the best set (generally with $N=[50\ 300]$, and $P=[6-60]$); using linear regression to train and test on the output of ICA. Surprisingly, neither smoothing the data nor using a ROI formed from a t-test of the movie on vs. off regions (which was considerably large) increased performance in our analysis.

More surprisingly, neither our sparse logistic regression algorithm nor our hybrid linear classifier-regression algorithm provided results that out-performed standard linear regression, given the parameter ranges we were able to explore. Theoretically, our sparse and hybrid algorithms should have provided results superior to linear regression because both use prior statistical information about the features being fitted to pick the voxels that provide the best

predictive information. However, standard linear regression allowed for a much more thorough examination of the parameter space, since it is considerably much less complex than both sparse regression and the hybrid algorithm. Therefore, given our time and computational constraints, we may not have sufficiently explored the parameter space of our sparse and hybrid algorithms. Thus, we submitted results primarily based on linear regression results, except for a few features in subject 1, for which some of the hybrid results were incorporated.

While the performance of our results relative to other approaches is unknown, we nevertheless feel that our approach provides a powerful method for localizing and predicting many high-level features. Given that linear regression (combined with various parameter-optimized correlation-based methods of preprocessing) provided better overall results than two prior-based methods (for which optimal preprocessing parameters could not be determined due to computational complexity), our results demonstrate that preprocessing techniques, particularly correlation-based techniques, contribute significantly to, if not more than, many regression techniques themselves towards feature prediction. Nevertheless, should sufficient time and computational power be available, a thorough examination of the parameters of various prior-based methods of regression and classification should provide one of the most powerful methods for predicting high-level features of brain imaging data in general.

References

Hyvärinen, A. and Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492.

Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634.

Sato, M., Yoshioka, T., Kajiwara, S., Toyama, K., Goda, N., Doya, K., Kawato, M. (2004). Hierarchical bayesian estimation for MEG inverse problem. *NeuroImage*, 23, 806-826.

Tipping, M.E. (2001). Sparse Bayesian Learning and the Relevance Vector Machine. *J. Machine Learning Research*, 1: 211-244.

Appendix Materials

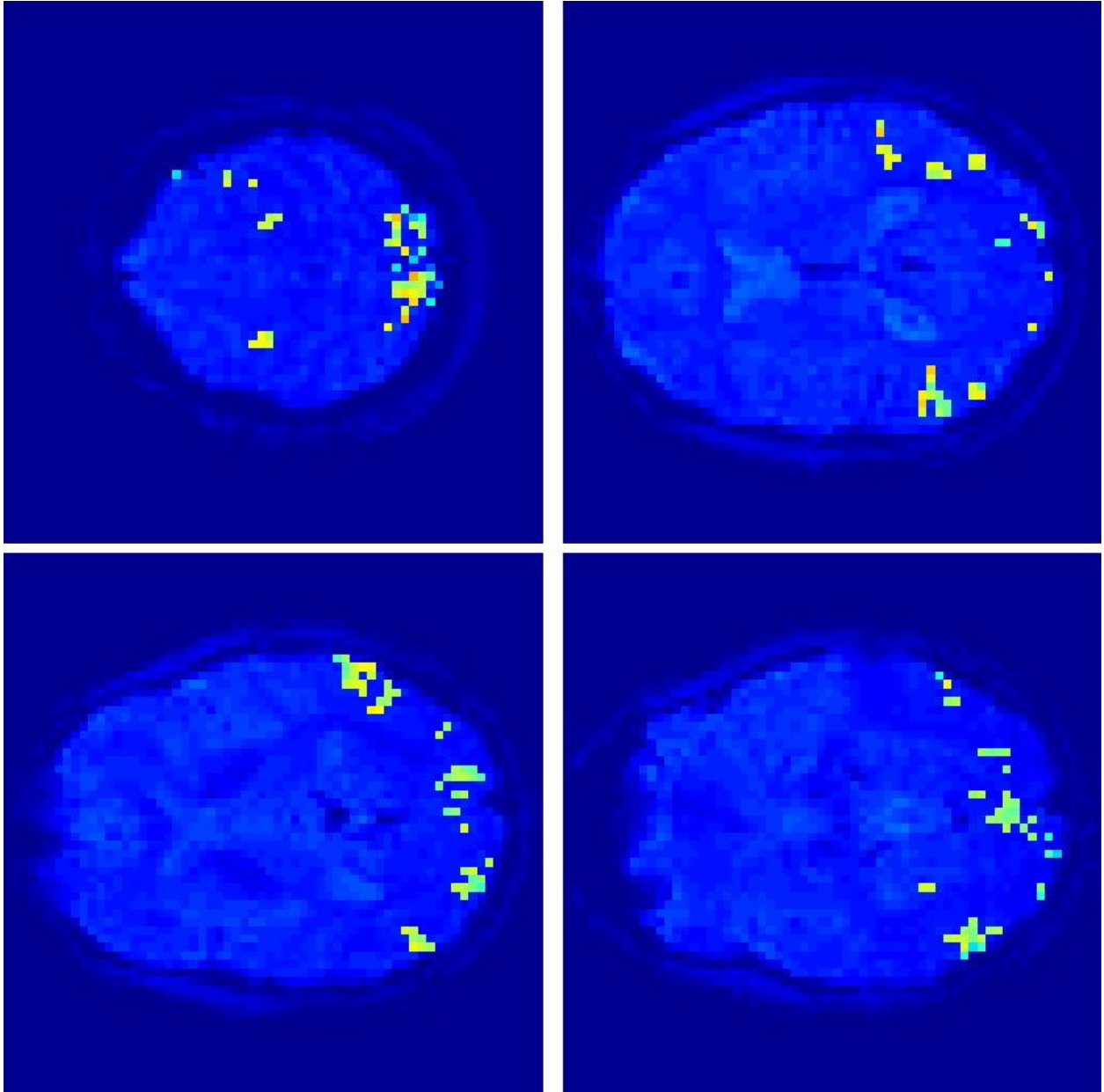


Figure 1: Voxels (green-red) maximally correlated with Motion superimposed on the first whole brain functional image (blue). Function slices 31, 19, 16, and 14 are shown from top to bottom, left to right.

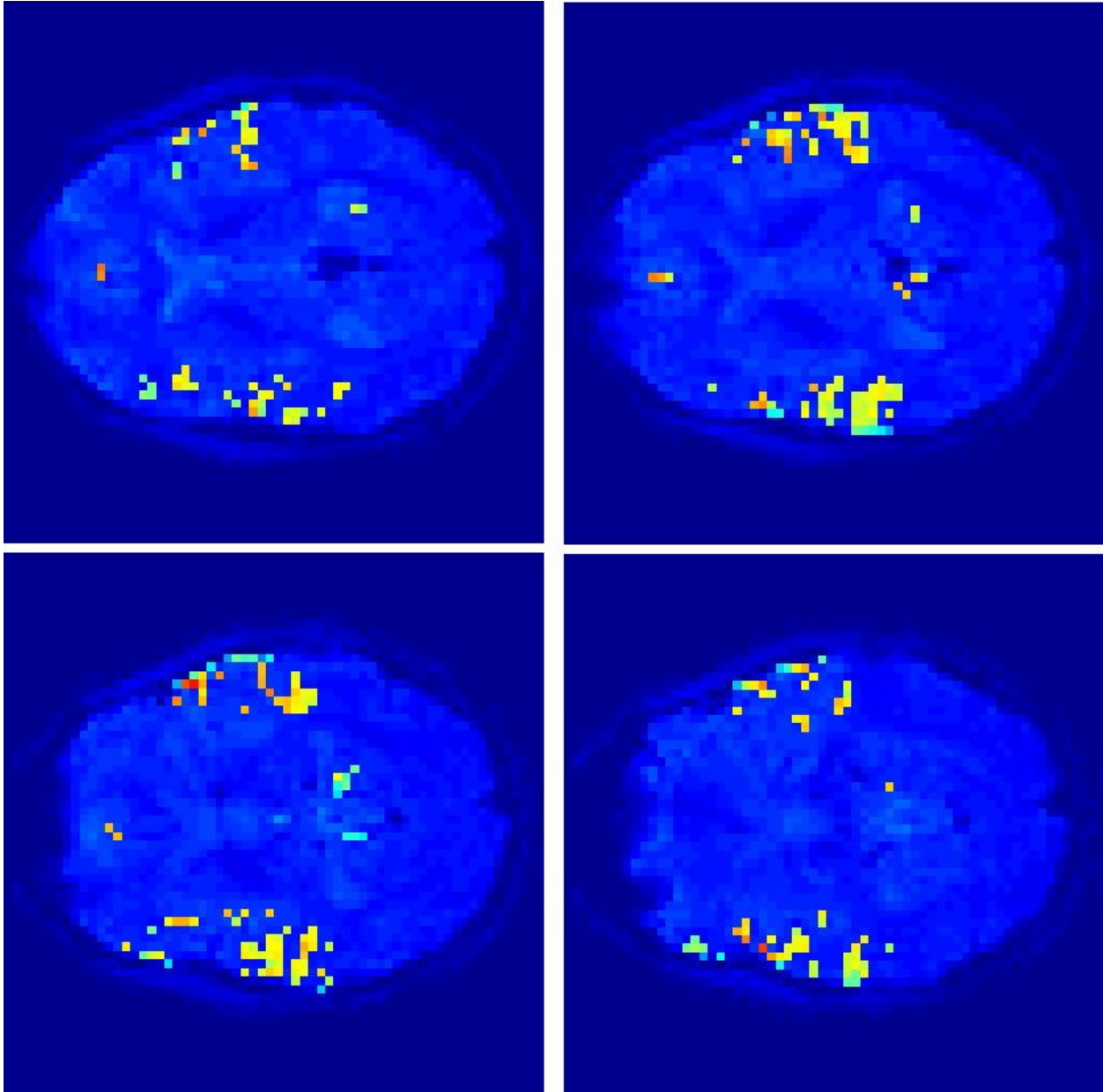


Figure 2: Voxels (green-red) maximally correlated with Language superimposed on the first whole brain functional image (blue). Function slices 17, 16, 15, and 14 are shown from top to bottom, left to right.

Optional Comment on Competition

1. Consider performing a couple of additional runs to localize and provide functional ROI's for some specific features, such as motion and language.
2. Provide all function ROI's in native-space, analyze format. Providing this only in Brain Voyager format created a slight bias of the competition toward using Brain Voyager, which is out of the price range of many participants.

3. Provide a co-registered, downsampled structural image that exactly matches the 64x64x32 functional image. Certainly, we can do this, in addition to the function ROI of movie-blank, but these are additional steps of unnecessary preprocessing that subtract from the competition's aim of participants examining difference decoding techniques.

4. Please provide more than 9 hours after the data submission deadline for the Methods Report deadline. There was no mention of a Methods Report deadline in earlier documentation, and it was only posted 4 days before the deadline itself. This resulted in both a great inconvenience for our group (we have a live television demo of our experiment tomorrow!) and in a relatively poor draft of this document I was forced to submit by the deadline. Since this document is more for the benefit of others, it seemed greatly counter-productive to require a strict report deadline immediately after the data submission deadline.

5. I strongly recommend using a discrete 1, 2, 3, 4 or 1, 2 rating system. Given the nature of the features and the descriptions given to the subject, using a continuous rating system is not only unnecessary but biases the feature data in favor of regression approaches instead of classifier approaches because it is much easier to convert discrete data into continuous data than convolved continuous data into discrete levels. Since classifiers usually provide much more powerful methods of prediction than regression methods, they should be the focus of the competition. However, it is very difficult for classifiers to produce continuous vectors that are more highly correlated with convolved continuous data than regression methods. For some feature data, consider 1 or 2 for present or absent.

6. In combination with (5), consider using a percent correct measure for discrete levels, and/or consider using the mean squared error (MSE) for more continuous data. MSE measures the lack of features more accurately than the correlation coefficient. Correlation metrics also put significantly more bias on features that are infrequent and high-level (e.g. 0.8-1 values), while feature levels that are closer to the mean have very little weight, and more common features have less weight per peak. Is not the level and lack of features also important to predict? And should more common features have more weight? Peaks can also throw off the mean, which also affects the correlation.

7. Shift all final vectors ± 2 secs and take the maximal score for each shift. In our decoding studies, we have observed that the hemodynamic delay varies. Since this is the case, one should have the freedom to shift the volume delay or hdf by ± 2 secs to optimize predictions. However, if this is done with your feature vectors, either on purpose or by accident, then the correlations could be extremely poor even though a ± 2 secs shift of the prediction versus empirical feature vectors before correlation could yield the winning results, and legitimately so, since ± 2 secs is still within the hdf. Such a shift may also occur as an artifact of up/down sampling of the feature vectors and/or volumes and might also result from confusion over optimizing the predictions for hdf or raw, when you might choose the opposite vector for judging the test data.