FMRI BRAIN IMAGING

Project Milestone, CSE P546, Winter 2014

Neeraj Prashar & Hari Sudan Sundar

Abstract

fMRI brain scan voxels are used to predict the cognitive state of the brain. In this particular effort, we have set out to explore how best to select and use the voxels as features to do the same. During the time till project milestone, we understood the dataset and ran Linear Regression and Logistic Regression experiments, which attempted to use all voxels as feature inputs for the regression and classification problems, respectively. The future direction of work that is planned for rest of the project is also summarized at the end.

Motivation

fMRI brain scan experiments require voluntary participation from the subject and thus the number of trials/samples are small. So, an inherent challenge with this problem is the high number of dimensions, when treating each voxel as a feature and relatively low number of observations, i.e. d >> n. This opens up a lot of interesting opportunities in trying out methods that can effectively model the voxel activity, taking care of the functional correlation between the different voxels and choosing the most influential voxel features to predict the cognitive state of the human brain.

Experiments

Categories

The training sample distribution was understood and it was concluded that it had a uniform distribution over the number of trials for each category of words. The following 12 categories were created according to the word's contextual meaning to enable us to split the training data uniformly to create a validation dataset and also for later use in multi-class classification using Logistic Regression with L1 penalty.

bear,cat,cow,dog,horse (animals)
arm,eye,foot,hand,leg (bodyparts)
apartment,barn,church,house,igloo (buildings)
arch,chimney,closet,door,window (buildingparts)
coat,dress,pants,shirt,skirt (clothing)
bed,chair,desk,dresser,table (furniture)
ant,bee,beetle,butterfly,fly (insects)
bottle,cup,glass,knife,spoon (kitchen utensils)
bell,key,refrigerator,telephone,watch (appliances)
chisel,hammer,pliers,saw,screwdriver (tools)
carrot,celery,corn,lettuce,tomato (vegetables)
airplane,bicycle,car,train,truck (transport vehicle)

Setup

We set apart a set of 60 training samples out of 300, to act as the validation dataset. We picked this validation dataset also to be a uniform distribution over the categories listed above. So, we worked with 240 training samples and used the rest 60 as validation dataset. All experiments were conducted on the validation dataset to enable us try different values for the parameters of the model (the regularization strength for both linear and logistic regressions) and chose that value of lambda that gave the least mistakes when tried against the validation dataset. Mistakes calculation for each experiment (Linear and Logistic Regression) are done a little differently and explained in the corresponding sections.

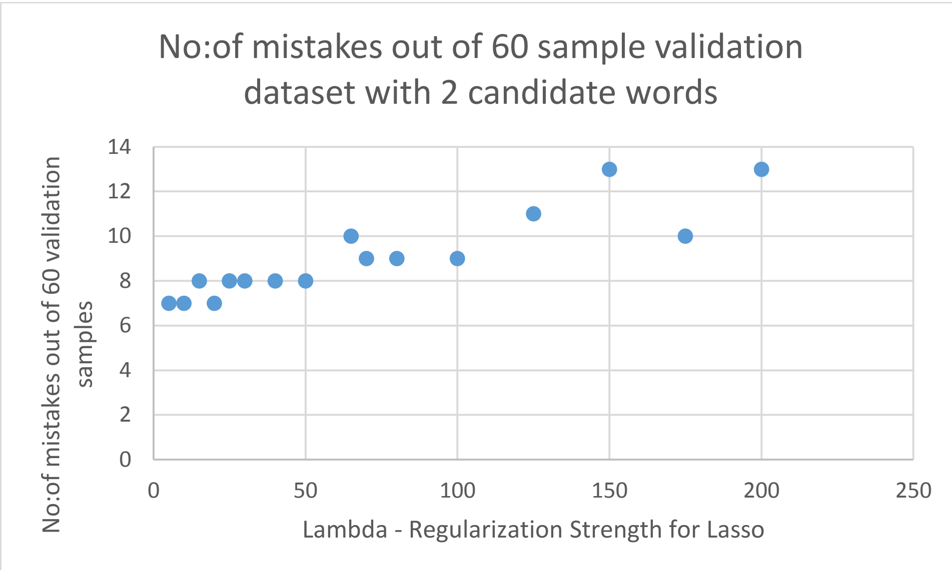
L1 Regularized Linear Regression (Lasso)

Since the number of dimensions **d >> n** (number of samples), we set out to get a sparse set of voxels that can be used to predict the 218 intermediate semantic feature vectors. So, we chose LASSO for its nature of giving sparse solutions.

1. We prepared a set of random words for each of the 60 validation dataset words, to act as a candidate word along with the truth word.
2. We trained 218 LASSO regression models to arrive at a predicted value for each of the 218 intermediate semantic features.

3. We calculate the cosine similarity between this 218 dimension semantic vector that the model predicted and the semantic feature vector of each of the 2 candidate words (the truth word and a random word – they both can't be same) – we chose that word as the prediction with whom the model vector shared more similarity.
4. Mistake Calculation – If the above process chose the truth word, it is not a mistake. If it chose the random word, it is counted as a mistake.
5. We trained such 218 LASSO regression models for each value of lambda (regularization strength) and used each lambda's prediction against the validation dataset, to choose the best lambda, for later use against the test dataset.

   Following plot shows the number of mistakes on validation dataset against different values of lambda.



Results against Test Set

So, we chose the lambda that gave the best results against the validation set (7 mistakes for lambda value of 10) and chose that to run our model against the actual test dataset, training on the whole 300 strong training dataset.

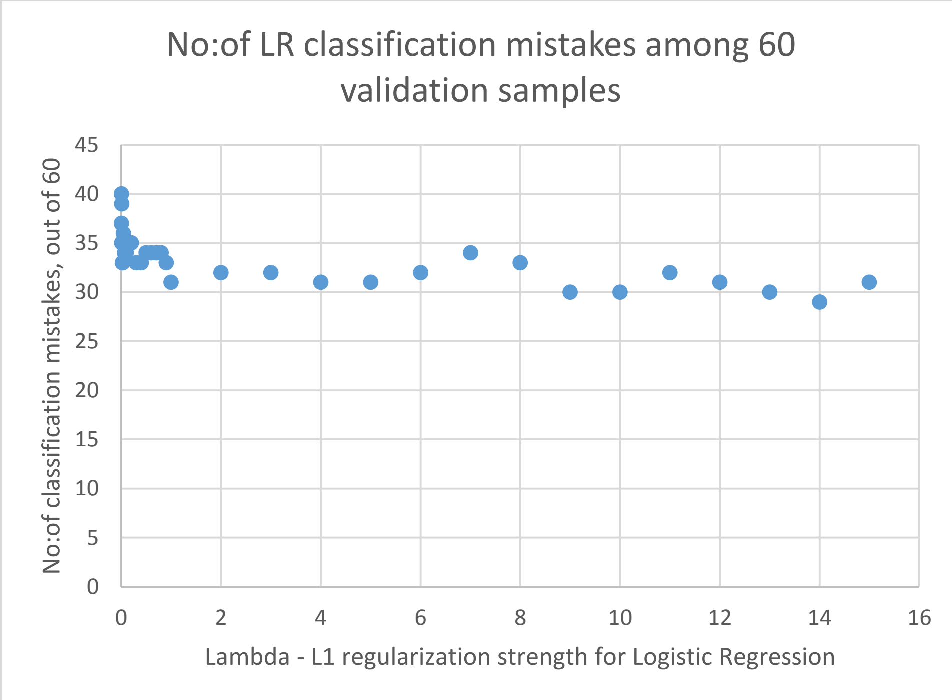
The number of mistakes when choosing between two candidate words were **8 out of 60,** on the test set.

Multi-class classification using Regularized Logistic Regression

While the intermediate representation as a vector of semantic features made sense in the context of solving this as a regression problem, we attempted to explore this problem from a different perspective, i.e., as a classification of the word being shown to the subject in to one of many categories we had created (12 of them). Here, we tend to ignore the intermediate semantic feature representation and not confine ourselves to just solving the binary classification problem of choosing between two candidate words. We wanted to let the classifier choose **among all the possible classes**, the best cognitive state this voxel activity could possibly represent. We divided the set of words shown to the subject in to 12 classes, based on the contextual meaning of the word, as mentioned before.

For logistic regression, since d >> n, we chose to use L1 regularization to create sparse solutions.

Following is the plot which shows the number of classification mistakes out of 60 validation samples against different values of the regularization strength – lambda.



Results against Test Set

From the above results, we pick the lambda that resulted in the least number of mistakes – which happens to be the value of lambda = 14, which yielded 29/60 mistakes on validation set. With this lambda, the number of classification mistakes on test set came out to be 30/60, which barely meets the chance bar. Using L2 regularization resulted in 29/60 mistakes.

Looking Forward

While we continue to analyze how we could better model this as multi-class classification – we plan to respect the following characteristic of the problem:

> Voxel activity is arbitrary with regard to underlying neural activity – and there is expected to be high correlation between nearby/related voxels.

Our goal for the rest of the project would be to come up with methods to generate highly influential features that better capture the underlying neural activity.

Some of the areas planned to be explored are:

1. Dimensionality reduction techniques, like the PCA
2. The influence of Spatial-temporal correlation between voxels
3. Fused lasso techniques to create sparsity of highly correlated voxel features.
4. Region of Interest (ROI) based correlation techniques which attempts to study the neural activity by dividing brain in to various regions of interest.

Code, Results:

For more information/details around the code behind our experiments, results, etc., please refer to our shared GIT repository https://github.com/prashar/FMRIClassifier. Parts of code also submitted in dropbox - Classifier_candidatewords.py and logistic_reg.py are the main files.