Multivariate Analysis of fMRI using Fast Simultaneous Training of Generalized Linear Models (FaSTGLZ)

Bryan Conroy BC2468@COLUMBIA.EDU
Paul Sajda PSAJDA@COLUMBIA.EDU
Columbia University, New York, NY

Abstract

We present an efficient algorithm for simultaneously training elastic-net-regularized generalized linear models across many related problems, which may arise from bootstrapping, cross-validation and nonparametric permutation testing. Our approach leverages the redundancies across problems to obtain $\approx$ 10x computational improvements relative to solving the problems sequentially by the standard glmnet algorithm of (Friedman et al., 2010). We demonstrate our fast simultaneous training of generalized linear models (FaSTGLZ) algorithm, for multivariate analysis of fMRI and run otherwise computationally intensive bootstrapping and permutation test analyses that are typically necessary for obtaining statistically rigorous classification results and meaningful interpretation.

1. Introduction

In functional MRI (fMRI) studies of the human brain, multivariate pattern analysis (Norman et al., 2006) has been shown to be a powerful technique for aggregating activity across spatially-distributed brain regions to predict various markers of cognitive state. Coupled with this goal is the desire to make inferences about the workings of the brain and its underlying cognitive processes. For this reason, a wave of attention has been recently focused on developing models that are both parsimonious and interpretable.

Sparse regression models, for example, have shown the potential to both obtain meaningful predictive models of brain state and also identify the relevant regions involved in the processing of a particular task or stimulus (Carroll et al., 2009). Despite their power, their use has thus far been limited by the computational burden involved in obtaining measures of statistical significance of prediction accuracy and parameter estimates. These operations typically involve re-training the model many times on different versions of the data.

We address this limitation by presenting an efficient algorithm to simultaneously train sparse regression models across many related problems. These problems may arise from bootstrapping, cross-validation, and permutation testing. We show that by solving the set of problems as a group, we are capable of leveraging the shared structure to obtain significant computational savings. Our algorithm applies to generalized linear models that are regularized by the elastic net. Generalized linear models (GLZ) are very flexible in that they are compatible with many popular probability distributions, such as the Gaussian, binomial, Poisson, exponential, and Laplace distributions. Under the binomial distribution, for example, the GLZ is equivalent to logistic regression. The elastic net regularization allows for sparse and parsimonious solutions while avoiding the saturation problems when the number of features p exceeds the number of examples n (Zuo & Hastie, 2005). This is often the case in fMRI studies: the number of acquired brain voxels can exceed the number of trials by an order of magnitude or more.

Below we begin with preliminaries of the problem definition and notation. We then describe our algorithm for fast simultaneous training of GLZ, which we term FaSTGLZ. We present results demonstrating the computational efficiency of FaSTGLZ, along with classification performance and functional neuroanatomical interpretability on an event related fMRI dataset.

2. Preliminaries

We start with a dataset $\left\{(x^{(i)}, y^{(i)})\right\}_{i=1}^{n}$, with features $x^{(i)} \in \mathbb{R}^p$ and response $y^{(i)}$. In the context of fMRI, the features may represent the BOLD responses from brain voxels, while y may indicate the category of the presented stimulus. For convenience, the feature data will be assembled into a $p \times n$ data matrix X.

GLZ's assume that the distribution of y is a member of the exponential family of distributions:

$$p(y|\theta,\phi) = c(y,\phi)\exp\left((y\theta - b(\theta))/a(\phi)\right) \tag{1}$$

where θ is the natural parameter, ϕ is a dispersion parameter, and a, b, c are known functions. The dependence between y and the feature data x arises by a link function that relates the mean of y to x. This is often accomplished by replacing the natural parameter θ in (1) with a linear predictor $x^T w$, where $w \in \mathbb{R}^p$ weights the relative importance of the features. In doing so, the mean of y, written $\mu(w)$, and the negative log-likelihood $\ell(w)$ are given by:

$$\begin{aligned} \mu(w) &= b'(x^T w) && (2)\\ \ell(w) &= -\sum_{i=1}^{n} d^{(i)}\left[y^{(i)}\eta^{(i)}(w) - b(\eta^{(i)}(w))\right] && (3)\end{aligned}$$

where $d, \eta(w) \in \mathbb{R}^n$ with $d^{(i)} = 1/a(\phi^{(i)})$ and $\eta(w) = X^T w$ is a vector of linear predictors. We will assume that the dispersion parameters $\phi^{(1)}, \ldots, \phi^{(n)}$ are known and in most cases, $d^{(i)}$ will be set to $1/n$.

The elastic net regularizes $\ell(w)$ by a mixture of ℓ_1 and ℓ_2 norms, so that our objective is given by:

$$\min_w J(w) = \min_w \ell(w) + \lambda_1||w||_1 + \lambda_2||w||_2^2 \tag{4}$$

where $\lambda_1, \lambda_2 \geq 0$ are tuning parameters that trade-off sparsity and smoothness. This is a convex optimization problem, for which many efficient algorithms have been proposed, e.g. (Friedman et al., 2010).

Our goal is to solve a multitude of such problems simultaneously. Since each problem will generally optimize $J(w)$ with respect to a distinct version of the data, each will have its own log-likelihood term $\ell_k(w_k)$, where w_k represents the unknown weights for problem $k \in \{1, \ldots, K\}$. For clarity, we use a subscript k on a variable to emphasize that it is specific to the k^{th} problem. The allowable variability in $\ell_k(w_k)$ across problems may be expressed by introducing problem-specific d_k and y_k, so that (3) is adapted to:

$$\ell_k(w_k) = -\sum_{i=1}^{n} d_k^{(i)}\left[y_k^{(i)}\eta^{(i)}(w_k) - b(\eta^{(i)}(w_k))\right] \tag{5}$$

Cross-validation, bootstrapping, and nonparametric significance testing all fall under this framework. For example, let $F_k = [f_{k1}, \ldots, f_{kn}]$ denote the relative frequencies of the training examples derived from a bootstrap or cross-validation fold. Its log-likelihood $\ell_k(w_k)$ may be expressed in the form of (5) by setting $d_k^{(i)}$ to f_{ki}. Note that if an index j is excluded (e.g., a sample belonging to the validation set of a cross-validation fold), then $d_k^{(j)} = 0$ and the j^{th} sample does not exert any influence on the objective.

Significance testing by nonparametric permutation testing also fits the form of (5). Here, the GLZ is re-trained on new datasets in which the response y has been permuted across examples. In this case, each problem k is given its own y_k, which is a permutation of the original sample $y^{(1)}, \ldots, y^{(n)}$.

To summarize, our goal is to minimize the regularized objectives $J_k(w_k)$, $k = 1, \ldots, K$:

$$\min_{w_k} J_k(w_k) = \min_{w_k} \ell_k(w_k) + \lambda_1||w_k||_1 + \lambda_2||w_k||_2^2 \tag{6}$$

Under cross-validation and bootstrapping, the variability in ℓ_k arises through problem-specific d_k, while permutation testing utilizes distinct y_k.

3. Methods

Our approach rests on the optimization procedure of alternating direction method of multipliers (ADMM) (Eckstein & Bertsekas, 1992). For each problem $k = 1, \ldots, K$, we divide the objective function $J_k(w_k)$ in (6) into the sum of two terms: the differentiable portion $f_k(w_k) = \ell_k(w_k) + \lambda_2||w_k||^2$, and the non-differentiable ℓ_1 term $g(w_k) = \lambda_1||w_k||_1$. An equivalent optimization problem is then formulated through the introduction of an auxiliary variable $v_k \in \mathbb{R}^p$:

$$\begin{gathered}\min_{w_k, v_k} \ell_k(w_k) + \lambda_2||w_k||^2 + \lambda_1||v_k||_1 \\ \text{subject to } w_k = v_k\end{gathered}$$

whose augmented Lagrangian may be expressed as:

$$\mathcal{L}_k(w_k, v_k) = f_k(w_k) + g(v_k) - \lambda_k^T(w_k - v_k) + \frac{1}{2\mu}||w_k - v_k||^2 \tag{7}$$

where $\lambda_k \in \mathbb{R}^p$ are estimates of the Lagrange multipliers and $\mu \geq 0$ is a penalty parameter.

Optimization proceeds by alternating between minimizing (7) with respect to w_k while holding v_k fixed, and vice versa. The resulting subproblems are substantially simpler than the original: optimizing w_k involves a differentiable objective, while updating v_k reduces to a soft-thresholding operation. The Lagrange multiplier estimates are updated after each of the above steps by $\lambda_k \leftarrow \lambda_k - (1/\mu)(w_k - v_k)$.

For our purposes, it is more convenient to re-parameterize the algorithmic steps in terms of a variable l_k, which is related to the Lagrange multiplier estimates λ_k. Specifically, given initial values for

w_k, v_k, λ_k, we initialize l_k to $l_k = \lambda_k + (1/\mu)v_k$. Then the ADMM procedure is equivalent to:

$$\begin{aligned} w_k &\leftarrow \arg\min_{w_k} S_k(w_k, l_k) && (8)\\ l_k &\leftarrow l_k - (2/\mu)w_k && (9)\\ v_k &\leftarrow -\mu \operatorname{soft}(l_k, \lambda_1 \mathbf{1}) && (10)\\ l_k &\leftarrow l_k + (2/\mu)v_k && (11) \end{aligned}$$

where

$$S_k(w_k, l_k) = \ell_k(w_k) + \rho||w_k||^2 - l_k^T w_k \quad (12)$$

and $\rho = \lambda_2 + \frac{1}{2\mu}$. Also, $\operatorname{soft}(a,b) = \operatorname{sgn}(a)\max(|a| - b, 0)$ is the soft-thresholding operator.

The effectiveness of our approach hinges upon efficiently solving (8). Here we show that it may be efficiently minimized simultaneously across all K problems using a Newton-type method. Specifically, we sequentially minimize a quadratic approximation to S_k:

$$\min_{w_k} q_k(w_k, \bar{w}_k) + \rho||w_k||^2 - l_k^T w_k \quad (13)$$

where $q_k(w_k, \bar{w}_k)$ is a quadratic approximation to $\ell_k(w_k)$ around an initial guess $\bar{w}_k$. Ignoring terms that don't depend on w_k, this may be written as:

$$q_k(w_k, \bar{w}_k) = \frac{1}{2} w_k^T H_k w_k + (\nabla \ell_k - H_k \bar{w}_k)^T w_k \quad (14)$$

where $\nabla \ell_k = X e_k$ and $H_k = X R_k X^T$ are the gradient and Hessian of $\ell_k(w_k)$ at $\bar{w}_k$, with $e_k = d_k \circ (\mu(\bar{w}_k) - y_k)$ and R_k is a diagonal matrix of positive values.

Thus, the minimizer of (13) may be solved in closed-form by solving the linear system of equations:

$$(H_k + 2\rho I)w_k = H_k \bar{w}_k - \nabla \ell_k + l_k \quad (15)$$

Inverting this linear system would typically be prohibitive since p is large. However, we can avoid this problem by exploiting the discrepancy between p and n. Let $X = QZ$ be a low-rank QR-decomposition of the data matrix, with $Q \in \mathbb{R}^{p \times n}$ a matrix with orthonormal columns and $Z \in \mathbb{R}^{n \times n}$, and denote P_Q as the projection onto the subspace spanned by the columns of Q. Then it can be proven that the solution to (15) may be expressed as:

$$w_k = Q\alpha_k + \frac{1}{2\rho}(I - P_Q)l_k \quad (16)$$

where $\alpha_k \in \mathbb{R}^n$ satisfies:

$$(Z R_k Z^T + 2\rho I)\alpha_k = Z(R_k Z^T \bar{w}_k - e_k) + Q^T l_k \quad (17)$$

This has reduced the problem to inverting a system of $n \ll p$ equations. Moreover, we can simultaneously solve (17) across all K problems very efficiently using the algorithm described in (Conroy & Sajda, 2012). Even though R_k in (17) varies across the K problems, only one matrix inversion is required to solve (17) for all $k = 1, \ldots, K$. Due to space constraints, we refer the reader to (Conroy & Sajda, 2012) for more details.

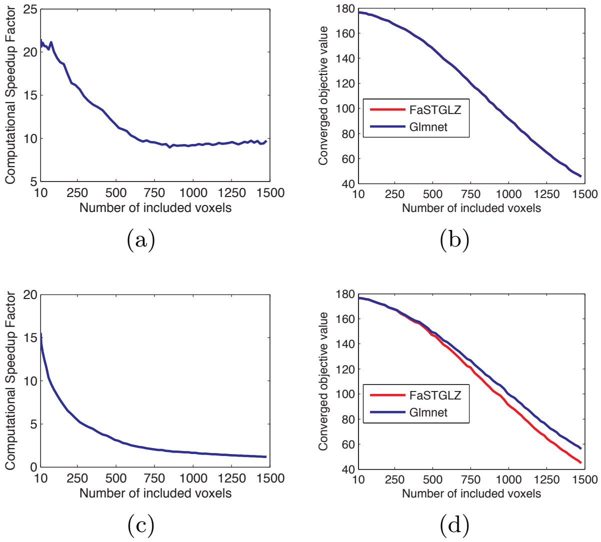

Figure 1. Benchmarking our algorithm against glmnet.

Finally, we utilize the screening rules for ℓ_1-regularized problems to determine the active set of each problem (Tibshirani et al., 2012).Without this, we would need to store l_k, w_k, v_k in full for each k, with $O(pK)$ elements. Since p is often very large, this would limit the number of problems K that could be solved simultaneously. Instead, by having an estimate of the active set A_k for each k, our memory overhead is reduced to $O(nK + s)$, where $s = \sum_{k=1}^{K} |A_k|$. Since $n \ll p$, this is more scalable for moderate sparsity levels.

4. Results

We analyzed our algorithm on a classification problem of predicting the stimulus category from functional MRI (fMRI) data collected during an auditory oddball detection task (Goldman et al., 2009). There are two categories (oddball/standard), so the GLZ is equivalent to logistic regression. For each of 3 subjects, $n = 374$ trials were acquired, and features corresponded to the BOLD response from brain voxels, with $p \approx 42{,}000$.

First, we benchmarked the speed of our algorithm in solving a set of K problems against solving them sequentially using the popular glmnet algorithm using coordinate descent (Friedman et al., 2010). Following (Friedman et al., 2010), we parameterized the regularization parameters (λ_1, λ_2) in terms of $(\alpha\lambda, 0.5(1 -$

Table 1. Maximum z-score per cluster (and cluster size)

Subject 1	Size of active set				
ROI	100	250	500	750	1000
F. Orbital	2.0(2)	2.6(6)	2.7(11)	2.6(10)	2.5(8)
Cingulate	-	-	1.7(1)	1.8(2)	1.9(3)
Heschl's Gyr.	-	-	-	1.7(2)	1.8(2)
Subject 2	Size of active set				
ROI	100	250	500	750	1000
R. Thalamus	2.8(18)	2.9(17)	2.4(8)	2.0(3)	1.7(1)
Heschl's Gyr.	-	-	1.8(2)	1.8(4)	1.8(9)
C. Opercular	-	-	-1.8(2)	-1.8(2)	-1.7(2)
Subject 3	Size of active set				
ROI	100	250	500	750	1000
L.Thalamus	2.1(5)	2.5(13)	1.9(5)	-	-
Cerebellum	-	-2.1(8)	-2.8(28)	-2.9(31)	-3.0(30)
Cingulate	-	2.0(2)	2.3(4)	2.0(2)	1.8(2)

$\alpha)\lambda)$ and held $\alpha = 0.7$ fixed, while λ varied along a regularization path of 100 values. As an example of a significance testing problem, we trained the classifier along this regularization path for $K = 1000$ permutations, and compared the time required by the two algorithms. Figure 1(a) plots the computational speedup factor, defined as the ratio of time required by glmnet to the time required by our algorithm, as a function of the average number of voxels included in the model. Our algorithm is at a minimum 9x faster. We verified that the relative difference in the converged objective value between the two algorithms never exceeded 2×10^{-4}, and a plot of the converged objectives is shown in Figure 1(b). As a further check, we also increased the convergence tolerance on glmnet so that it ran the full regularization path in roughly the same time as our algorithm (our algorithm was still 1.2x faster – see Figure 1(c)). In this case, (see Figure 1(d)), our algorithm produced a converged objective value that was approximately 20% lower than glmnet.

As a further application of our algorithm, we also ran a joint cross-validation and bootstrapping analysis to obtain both cross-validated classification accuracy and significance levels on the derived classifier weights. For each of the 3 subjects, we used 10-fold cross-validation to obtain prediction accuracy, and within each fold, we re-trained on 100 bootstrap samples of the training set. The bootstrap resampling was used to convert the classifier weights to z-scores. We then identified spatial clusters that exhibited significant z-scores ($|z| \geq 1.64$). Tables 1 lists the maximum z-score and size of the clusters found from this analysis. There is some consistency in the selected regions, as multiple subjects draw from Heschl's gyrus, Cingulate gyrus, and the thalamus.

The corresponding area under the ROC curve (Az) values for each of the classifiers are given in Table 2. To verify prediction accuracy significance, we ran a permutation test using our algorithm, and obtained a significance level of $Az = 0.64$, $p < 0.01$.

Table 2. Area under the ROC curve (Az)

	Size of active set				
Subject	100	250	500	750	1000
Subject 1	0.73	0.77	0.78	0.77	0.76
Subject 2	0.80	0.84	0.85	0.86	0.86
Subject 3	0.87	0.91	0.93	0.94	0.94

5. Conclusion

We presented the fast simultaneous training of generalized linear model (FaSTGLZ) algorithm and demonstrated its ≈ 10x speedup in computational efficiency when performing analysis of large multivariate fMRI datasets. FaSTGLZ enables efficient implementation of any elastic-net regularized GLZ. Future work will investigate applying FaSTGLZ to link activity across neuroimaging modalities, as might be done with simultaneous EEG and fMRI.

Acknowledgements

This work was supported by a NIH grant R01- MH085092. We would like to thank Jennifer Walz for use of the 3T fMRI auditory oddball data.

References

Carroll, M.K., Cecchi, G.A., Rish, I., Garg, R., and Rao, A.R. Prediction and interpretation of distributed neural activity with sparse models. *NeuroImage*, 44(1):112–122, 2009.

Conroy, B. and Sajda, P. Fast, exact model selection and permutation testing for ℓ_2-regularizaed logistic regression. In Lawrence, N. and Girolami, M. (eds.), *Proc. 15^{th} Intl Conf. Art. Intell. Stat.*, 2012.

Eckstein, J. and Bertsekas, D. On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 5:293–318, 1992.

Friedman, J., Hastie, T., and Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Soft.*, 33(1):1–22, 2010.

Goldman, R.I., Wei, C.Y., Philiastides, M.G., Gerson, A.D., Friedman, D., Brown, T.R., and Sajda, P. Single-trial discrimination for integrating simultaneous eeg and fmri: Identifying cortical areas contributing to trial-to-trial variability in the auditory oddball task. *NeuroImage*, 47(1):136–147, 2009.

Norman, K.A., Polyn, S.M., Detre, G.J., and Haxby, J.V. Beyond mind-reading: multi-voxel pattern analysis of fmri data. *Trends in Cog. Sci.*, 10(9):424–430, 2006.

Tibshirani, R., Bien, J., Friedman, J., and Hastie, T. Strong rules for discarding predictors in lasso-type problems. *J.R. Statist. Soc. B*, 74:245–266, 2012.

Zuo, H. and Hastie, T. Regularization and variable selection via the elastic net. *J.R. Statist. Soc. B*, 67:301–320, 2005.