Regularization and Feature Selection for Networked Features

Hongliang Fei, Brian Quanz, Jun Huan
Department of Electrical Engineering and Computer Science
University of Kansas
Lawrence, KS 66047-7621, USA
{hfei, bquanz, jhuan}@ittc.ku.edu

ABSTRACT

In the standard formalization of supervised learning problems, a datum is represented as a vector of features without prior knowledge about relationships among features. However, for many real world problems, we have such prior knowledge about structure relationships among features. For instance, in Microarray analysis where the genes are features, the genes form biological pathways. Such prior knowledge should be incorporated to build a more accurate and interpretable model, especially in applications with high dimensionality and low sample sizes. Towards an efficient incorporation of the structure relationships, we have designed a classification model where we use an undirected graph to capture the relationship of features. In our method, we combine both L_1 norm and Laplacian based L_2 norm regularization with logistic regression. In this approach, we enforce model sparsity and smoothness among features to identify a small subset of grouped features. We have derived efficient optimization algorithms based on coordinate decent for the new formulation. Using comprehensive experimental study, we have demonstrated the effectiveness of the proposed learning methods.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications-Data Mining

General Terms

Algorithms, Experimentation

Keywords

Regularization, Feature Selection, Logistic Regression

1. INTRODUCTION

Data whose features have intrinsic structure relationships is becoming abundant in many application domains such as bioinformatics, text mining, and computer vision, among others. For instance, in microarray classification, the genes form a biological network (the pathway graph) [7, 8]. In text mining where key words are features, we have additional information about synonyms or antonyms of the features. Such information is usually captured with a word net [1]. Exploring the intrinsic structure of features and utilizing such structure for deriving low dimensional representations of the original data is a critical step for accurate modeling and better model interpretation.

Feature selection for data with “structured features”, usually high-dimensional data whose features have a known and fixed structure, has recently attracted research intensive interest in the machine learning and data mining communities [11, 12, 13]. For example Tibshirani [11] studied feature selection for data whose features form a linear chain and proposed a method called fused lasso. Yuan and Lin explored the situation where features may be naturally partitioned into groups and devised a technique called grouped Lasso [12]. In [13], both group structure and hierarchical relation of features have been studied in a unified framework.

Adopting existing techniques to aforementioned applications where features adopts a graph structure, rather than groups, chains, or trees for classification, is non-trivial. The challenge originates from the optimization. For linear regression, it is relatively easier to manipulate least square loss function than hinge loss, logit loss and exponential loss that are widely used in classification. Furthermore, there is no natural way to incorporate the additional feature graph information into the classification framework. As a preliminary study, Li *et al.* utilized the pathway information of Microarray and proposed a gene selection method by combining L_1 norm and L_2 norm regularized pathway graph Laplacian [7]. However, their algorithm is constrained in linear regression for genomic data analysis. In [10], a networked feature regularization for classification was proposed, but it only imposed smoothness by L_2 norm graph Laplacian on the features and no feature selection was performed. Hence the method may not provide the optimal performance and we will demonstrate that in our experiment study.

In this paper, we devised a more general framework to perform graph structured feature selection for classification. We extended the coordinate descent algorithm presented in [2] for the Elastic Net to derive efficient optimization algorithms for a penalized logistic regression model with both L_1 norm and Laplacian based L_2 norm penalty. The advantage of the penalization framework is that it combines both the L_1 norm and the graph Laplacian penalties to enforce model sparsity and smoothness in the feature space. Using comprehensive experimental study, we have demonstrated the effectiveness of the proposed learning method.

The rest of the article is organized as follows. Section 2 discusses related work. Section 3 presents background information and detailed discussion of our algorithms. Sec-

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.

tion 4 presents the experimental study of our algorithms as compared to competing methods. Finally we give a short conclusion.

2. RELATED WORK

To handle structure prior among features, methods that are closely related to ours are fused lasso [11], network constrained regression [7] and logistic regression with with networks of features [10]. Fused lasso [11] can be interpreted as a regularization method with an L_1 norm penalty on the graph Laplacian, where the graph is a chain. However, the L_1 penalty in fused lasso forces the weights of neighboring features to be identical rather than similar, which is favorable in only a few applications. Li and Li [7] regularize feature weights by penalizing the normalized graph Laplacian for biomedical prediction tasks and their model only works for linear regression with least square loss function. Sandler *et. al* [10] proposed regularized learning on networks of features for logistic regression, in which the feature network is built from prior knowledge of whether the pairwise features are similar or dissimilar. However, they only introduce the L_2 norm penalty, therefore their model does not enjoy sparsity and cannot perform feature selection.

Our work is different from existing network regularized regression work in that we use a general graph to capture relationships between features for logistic regression. By incorporating both an L_1 penalty and a L_2 Laplacian penalty, we enforce model sparsity and smooth variation over the known graph, effectively selecting features that are grouped according to the known graph structure (see section 3 for details). Hence the key insight is that the lasso penalty introduces sparsity to the model and L_2 Laplacian penalty penalizes parameter values that diverge more from their neighboring parameters' values and obtains smoothness to achieve grouping effect. Though we exclusively work on binary classification in this paper, logistic regression naturally extends to multi-class classification and hence we do not expect any problems in applying our method to multi-class classification.

3. METHODOLOGY

3.1 Problem Statement

We consider a supervised learning problem with incorporation of structure prior among features for logistic regression. Given a set of n training samples $T = \{(\vec{x_i}, y_i)\}$, $\vec{x_i} \in \mathcal{X} \subset \mathbb{R}^p$, $y_i \in \mathcal{Y} = \{0, 1\}$, $i \in [1, n]$, the task of logistic regression is to learn model coefficients $\vec{\beta}$ so that the log-likelihood function (1)

$$\begin{aligned} loglik(\vec{\beta}) &= \textstyle\sum_{i=1}^{N}\{I(y_i = 1)\log p(\vec{x}_i) \\ &\quad + I(y_i = 0)\log(1 - p(\vec{x}_i))\} \\ &= \textstyle\sum_{i=1}^{N}\{y_i\vec{\beta}^T\vec{x}_i - \log(1 + \exp\vec{\beta}^T\vec{x}_i)\} \end{aligned} \tag{1}$$

is maximized, where $p(\vec{x}_i) = P(y_i = 1|\vec{x}_i) = 1/(1+exp(-\vec{x}_i^T\vec{\beta}))$ and $I(.)$ is the indicator function.

In our model, we capture the structure relationship among features as an undirected graph G, whose nodes correspond to the p features. Edges in the graph G are weighted, where the weight $w_{i,j} > 0$ indicates the "relationship" between the two features. We call this graph "feature graph". We incorporate the a priori domain knowledge by adding a Tikhonov regularization factor $\frac{1}{2}\sum_{i,j} w_{i,j}(\beta_i - \beta_j)^2$ in the convex fitness function $-loglik(.)$ to enforce smoothness for neighboring features. The Tikhonov regularization factor could be conveniently written in matrix format $\vec{\beta}^T L\vec{\beta}$ where L is the *Laplacian* of G given by: $L = D - W$. W is the p by p edge weight matrix $W = (w_{i,j})_{i,j=1}^p$, and D is the density matrix of W, defined as $D = (d_{i,j})_{i,j=1}^p$ where $d_{i,j} = \begin{cases} \sum_{k=1}^n W_{i,k} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$ To avoid having any feature "dominate" the penalization function, we use the *normalized Laplacian* $\mathcal{L} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$ to normalize the weight of each feature. Tikhonov regularization does not lead to the sparsity of the model. In other words, Tikhonov regularization cannot perform feature selection. To obtain a sparse model, we add the L_1 norm of $\vec{\beta}$ to the negative log-likelihood function. Specifically in our learning, we seek to identify a vector $\vec{\beta}$ that minimizes the following loss function: $\ell(X, \vec{y}; \vec{\beta}) = -loglik(\vec{\beta}) + \frac{1}{2}\lambda_2\vec{\beta}^T\mathcal{L}\vec{\beta} + \lambda_1||\vec{\beta}||_1$, where $\lambda_1 > 0$, $\lambda_2 > 0$, $||.||_1$ is L_1 norm. We restrict the problem to binary classification for simplicity. Logistic regression can be easily generalized to multi-class classification. A similar formalization was proposed in [7] for linear regression, however, their work focuses on linear regression for genomics study. We are investigating a more general classification problem rather than linear regression across a wide range of application domains.

Using L_2 norm regularized graph laplacian, our method provides two insights: 1) smoothness: the coefficients of neighboring features are close to each other due to the L_2 norm regularized feature graph Laplacian regularization. 2) Grouping effect: Once a feature is selected, its neighboring features will be more likely selected. Thus our algorithm can select groups of neighboring features in the feature graph.

3.2 Optimization Algorithms

We designed a coordinate descent [2, 3] algorithm to solve the logistic regression problem with the mixture penalty of L_1 and Laplacian regularized L_2 norm. In designing our optimization algorithm, we followed the general framework proposed in [3] for Elastic Net where both L_2 norm and L_1 norm are used for regularized linear regression. Adopting the solution for Elastic Net for our current problem is nontrivial. First, we use logistic regression, rather than linear regression. Second we use a Laplacian weighted L_2 norm $(\vec{\beta}^T\mathcal{L}\vec{\beta})$, rather than the regular L_2 norm $(\vec{\beta}^T\vec{\beta})$. Recently Friedman *et al.* [3] have proposed a quadratic approximation scheme for extending coordinate descent algorithm for logistic regression with L_1 penalty. Their strategy demonstrated computational efficiency and hence we adopt the same strategy to handle the first issue.

Below, we give a detailed derivation of a modified coordinate descent algorithm extending the work presented in Elastic Net for handling the combination of Laplacian regularized L_2 norm and L_1 norm to handle the second issue. In our derivation, we use the same least squares fitness function as Elastic Net and we discuss about the extension of replacing the least squares fitness function with the negative log-likelihood function later.

LEMMA 3.1. *Suppose that the data set contains n observations and p predictors, with the response vector $Y = (y_1 \ldots y_n)^T$ and the data matrix $X = (\vec{x}_1, \ldots, \vec{x}_n)^T$. We also assume that the predictors are standardized and the response is centered so that for all j, $\sum_{i=1}^n x_{ij} = 0$, $\sum_{i=1}^n x_{ij}^2 = 1$ and $\sum_{i=1}^n y_i = 0$. The Lagrange form of the objective function is $L(\lambda_1, \lambda_2, \vec{\beta}) = \frac{1}{2}(Y - X\vec{\beta})^T(Y - X\vec{\beta}) + \frac{1}{2}\lambda_2\vec{\beta}^T\mathcal{L}\vec{\beta} + \lambda_1||\vec{\beta}||_1$. The coordinate-wise update has the form (for each β_j): $\hat{\beta}_j = S(\sum_{i=1}^n x_{ij}(y_i - \tilde{y_i}^{(j)}) - \lambda_2\sum_{k\neq j}^p \mathcal{L}_{jk}\hat{\beta}_k, \lambda_1)/(1 + \lambda_2\mathcal{L}_{jj})$ where $\tilde{y_i}^{(j)} = \sum_{l\neq j} x_{il}\hat{\beta}_l$ is the fitted response value* *excluding the contribution from x_{ij} and $S(z, \gamma) = sgn(z)(|z| - \gamma)_+$ is the soft thresholding operator where:*

$$sgn(z)(|z| - \gamma)_+ = \begin{cases} z - \gamma & \textit{if } z > 0 \textit{ and } \gamma < |z| \\ z + \gamma & \textit{if } z < 0 \textit{ and } \gamma < |z| \\ 0 & \textit{if } \gamma \geq |z| \end{cases}$$

The derivation of Lemma 3.1 is based on coordinate descent for elastic net in [3]. Due to space constrain, we do not provide the details.

With Lemma 3.1, we have an effective solver for least square fitting with the mixture penalty of the L_1 norm and the regularized Laplacian L_2 norm. We followed the general framework of coordinated decent algorithm recently proposed by Friedman *et al.* [3] for L_1 norm regularized logistic regression. Their approach relies on the connection between the Newton's method for optimizing logistic regression and the least square formulation. The Newton's method amounts to using Taylor expansion, up to a quadratic function, to approximate the negative log-likelihood function. In this way, applying Newton's method can be viewed as solving a series of least squares problem (also called *iterative reweighted least squares fitting* [3]). Applying Taylor's expansion at current estimate $\tilde{\vec{\beta}}$ to negative log-likelihood function (1), we have the reweighted least square problem $l_Q(\vec{\beta}) = -\sum_{i=1}^{n} w_i(z_i - \vec{x}_i^T \vec{\beta})^2 + C(\tilde{\vec{\beta}})$ where $z_i = \vec{x}_i^T \tilde{\vec{\beta}} + (y_i^* - \tilde{p}(\vec{x}_i))/(\tilde{p}(\vec{x}_i)(1 - \tilde{p}(\vec{x}_i)))$, $w_i = \tilde{p}(\vec{x}_i)(1 - \tilde{p}(\vec{x}_i))$ and $C(\tilde{\vec{\beta}})$ is a constant. The Newton update is obtained by minimizing the following regularized reweighed least squares problem:

$$L_Q(\vec{\beta}) = -l_Q(\vec{\beta}) + \frac{1}{2}\lambda_2 \vec{\beta}^T \mathcal{L} \vec{\beta} + \lambda_1 ||\vec{\beta}||_1. \qquad (2)$$

It is obvious $L_Q(\vec{\beta})$ can be minimized by coordinate descent with the update rules proposed in Lemma 3.1.

We summarize what is briefly discussed previously in the algorithm called LogLapElasnet. Given the training data $\mathcal{X}$ and $\vec{y}$ and regularization parameters λ_1 and λ_2, our algorithm iteratively solves logistic regression with L_1 norm and L_2 norm penalty on normalized Laplacian of the feature graph.

Algorithm 1 LogLapElasnet($\lambda_1, \lambda_2, MaxIteration, \epsilon$)

```
Initialize $\hat{\vec{\beta}}^{(0)} = \vec{0}$;
for i=1 to MaxIteration do
    Compute the quadratic approximation for (1);
    Use the coordinate descent method in lemma 3.1 to
    solve the reweighted least squares problem (2) with
    mixture penalty and obtain the updated $\vec{\beta}^{(i)}$;
    if $||\hat{\vec{\beta}}^{(i)} - \hat{\vec{\beta}}^{(i-1)}||_1 \leq \epsilon$ then
        Break;
    end if
end for
return $\hat{\vec{\beta}} = \hat{\vec{\beta}}^{(i)}$;
```

4. EXPERIMENTAL STUDY

We have performed a rigorous evaluation of our regularized learning algorithm in terms of modeling accuracy and feature selection performance using two real-world data sets. For comparison, we implemented our own solver for Logistic regression with the Lasso (L_1) penalty (LogLasso), elastic net (L_2 plus L_1) penalty (LogElasnet), and the FNR [10] method with networked L_2 Laplacian penalty. We did not get a chance to compare with [5] but will study it in our future work.

4.1 Data sets

We used the following two data sets for our real-world data study:

Diabetes Data. The data set is obtained from [9] and contains the gene expression values of 22,280 genes for 44 different subjects, 17 with type 2 diabetes (DM2), 17 with normal glucose tolerance (NGT) and 10 with impaired glucose tolerance (IGT). As in [8], we use only the 34 samples of subjects with type 2 diabetes and those with normal glucose tolerance. We collected all pathway information from the Kyoto Encyclopedia of Genes and Genomes (KEGG) database [6] and use the global test method [4] to obtain related pathways to the diabetes outcome with a significance p-value of less than 0.1. We keep those 13 non-overlapping pathways and merge them to generate a graph.

20 Newsgroup Data. The data is available at `http://people.csail.mit.edu/jrennie/20Newsgroups/` and we use the second one (by date). We merge the original training (60%) and test (40%) sets to form a whole data set. To perform classification, we single out two classes that are very correlated to each other (comp.os.ms-windows.misc and comp.sys.ibm.pc.hardware). The feature set was constrained to be 610 key words which occur at least 25 times in these 1942 documents excluding stop words. Following the same procedure in [10], we build the graph on features. Refer to [10] for more information.

4.2 Evaluation Criteria

Below we present our approaches for model construction and model comparison. **Model Construction.** We partition the data set into 10-folds to perform 10-fold cross-validation (CV). We use another 10-fold CV on the training data set to select the regularization parameters λ_1 and λ_2 using a simple grid search. We then generate a single model from the entire training set with the selected parameters and apply the model to the testing data set for prediction.

Model Comparison. For model comparison, we collect the sensitivity (TP/(TP+FN)), specificity (TN/(TP+FP)) and accuracy ((TP+TN)/S) of the trained model, where TP stands for true positive, FP stands for false positive, TN stands for true negative, FN stands for false negative, and S stands for the total number of samples. All the values (specificity, sensitivity, accuracy) reported are collected from the testing data set only and are averaged across 10-fold CV with 3 replicates in a total of 30 experiments.

To compare different feature selection strategies, we also collect the selected features in each cross validation and report the average # features selected during the experiments. To demonstrate the group feature selection effect, for Microarray data where the feature graph is sparse, we collect the number of selected feature clusters or pathways. For News group data, the feature graph is dense and there is no natural way to partition the graph into "components". We define the *average feature separation* $\bar{d}$ as the average shortest path length of pairs of selected features: $\bar{d} = \sum_{i\ j} d(i, j)$ where $i, j \in F$, F is the selected features, $d(i, j)$ is the shortest path length between feature i and j in the original feature graph.

4.3 Classification Performance

In Table 1, we report the average test sensitivity, specificity, accuracy, number of selected features, number of selected pathways for the Microarray data, and the average feature separation for the NewsGroup data with 200 samples (randomly sampled 100 documents from the two classes) in the 3 replicates of 10-fold cross validation. As shown in Table 1, LogLasso selects the least number of features and LogElasnet selects the most. LogLapElasnet usually

Table 1: Average sensitivity (SEN), specificity (SPE), accuracy (ACC), number of selected features (F), the number of selected pathways for the Microarray data ($\mathcal{P}$), or the average feature separation for newsgroup data ($\overline{\mathcal{D}}$). Stars (*) denote the best value among all competing methods for a data set.

	Data Set:Diabetes					Data Set:News				
Methods	F	$\mathcal{P}$	SEN	SPE	ACC	F	$\overline{\mathcal{D}}$	SEN	SPE	ACC
LogLasso	9^*	9	0.57	0.58	0.64	23^*	308	0.68	0.84	0.72
LogElasnet	28	10	0.55	0.58	0.68^*	109	1159	0.92^*	0.66	0.74
LogLapElasnet	25	6^*	0.57^*	0.54	0.68	41	$146\ ^*$	0.75	0.88^*	0.78^*
FNR	701	13	0.47	0.63^*	0.61	610	8949	0.34	0.80	0.65

builds a relatively sparser model belonging to less number of pathways with comparable or even better classification performance and with a good balance between sensitivity and specificity of the model.

To further study the "grouping effect" for feature selection of our method for the Microarray data, we have singled out all genes that are selected in all 30 experiments and investigated the pathways that these selected genes belong to. We observe that genes selected by LogLapElasnet belongs to 9 pathways, while those selected by LogLasso belong to 12 pathways and 13 pathways for LogElasnet.

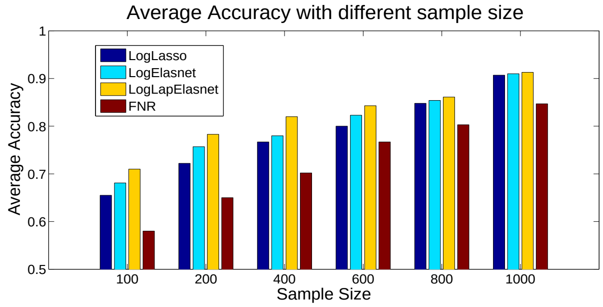

Figure 1: Average accuracy in 30 experiments for data sets with different sample size

4.4 Performance Gain with Small Sample Size Settings

We hypothesize that the regularization framework works best with small number of samples. To test the hypothesis, we randomly sampled equal number of instances from the two classes of newsgroup data and form 5 data sets with 100, 400, 600, 800 and 1000 samples. In Figure 1, we present the average accuracy for 10-fold CV with 3 replicates. First, there is a clear trend that the performance of all the methods increases as the number of samples is increasing. Second, the performance for the methods with feature selection is superior to that of FNR for all the cases. Finally, our method LogLapElasnet outperforms the others significantly when the sample size is low, and comparable to others when the sample size is high. Such facts confirm the effectiveness of our method for the problem of $n << p$ when exploring the structure information among features.

5. CONCLUSIONS

We present a learning framework of regularization and feature selection on networked features to incorporate prior knowledge of structure information for logistic regression. By introducing normalized graph Laplacian to regularization term, we combine L_1 and L_2 norm regularization to achieve both sparsity and smoothness with respect to the coefficients of features. Additionally, our method can select clustered features that are related to the outcomes. We have demonstrated the performance of our method on two real-world data sets.

Acknowledgments

This work has been supported by an Office of Naval Research award (N00014-07-1-1042) and the National Science Foundation under Grant No. 0845951.

6. REFERENCES

[1] C. Fellbaum. *WordNet: an electronic lexical database*. the MIT Press, 1998.

[2] J. Friedman, T. Hastie, H. Hofling, and R. Tibshirani. Pathwise coordinate optimazation. *The Annals of Applied Statistics*, 1:302–332, 2007.

[3] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *The Annals of Applied Statistics*, page to be appeared, 2009.

[4] J. J. Goeman, S. A. van de Geer, F. de Kort, and H. C. van Houwelingen. A global test for groups of genes: testing association with a clinical outcome. *Bioinformatics*, 20(1):93–99, 2004.

[5] L. Jacob, G. Obozinski, and J.P. Vert. Group lasso with overlap and graph lasso. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440, 2009.

[6] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, , and M. Hirakawa. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Research*, 34:D354–357, 2006.

[7] C. Li and H. Li. Newwork-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175–1182, 2008.

[8] L. Liang, V. Mandal, Y. Lu, and D. Kumar. Mcm-test: a fuzzy-set-theory-based approach to differential analysis of gene pathways. *BMC Bioinformatics*, 9(Suppl 6):S16, 2008.

[9] V. Mootha, C. Lindgren, K. Eriksson, A. Subramanian, S. Sihag, J. Lehar, P. Puigserver, E. Carlsson, M. Ridderstraale, E. Laurila, and et al. Pgc-1α-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics*, 34(3):267–273, 2003.

[10] T. Sandler, P. P. Talukdar, and L. H. Ungar. Regularized learning with networks of features. In *NIPS08*, 2008.

[11] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *J. R. Statist. Soc.*, 67(1)):91–108, 2005.

[12] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2006.

[13] P. Zhao and B. Yu. Grouped and hierarchical model selection through composite absolute penalties. *Annals of Statistics*, 2006.