GRADIENT SPARSE OPTIMIZATION VIA COMPETITIVE LEARNING

Nan Zhang, Shuqing Zeng and Juyang Weng

Michigan State University
Department of Computer Science and Engineering
East Lansing, MI, 48823
email:{nanzhang, zengshuq, weng}@cse.msu.edu

ABSTRACT

In this paper, we propose a new method to achieve sparseness via a competitive learning principle for the linear kernel regression and classification task. We form the duality of the LASSO criteria, and transfer an ℓ_1 norm minimization to an ℓ_∞ norm maximization problem. We introduce a novel solution derived from gradient descending, which links the sparse representation and the competitive learning scheme. This framework is applicable to a variety of problems, such as regression, classification, feature selection, and data clustering.

1. INTRODUCTION

The central problem of supervised learning or regression can be formulated as function approximation. In either case we have pairwise correspondence of samples $\mathbf{x}$ and $\mathbf{y}$ from two sample space $\mathbf{X}$ and $\mathbf{Y}$, and the task is to find a function $f(\cdot)$, such that $\mathbf{y} = f(\mathbf{x})$. More precisely, if the model of the function is chosen, the function can be written as $\mathbf{y} = f(\mathbf{x}, \beta)$, where β is the parameter vector of the model. For example, the linear kernel regression assumes such function is a linear combination of a set of basis functions, i.e. $\mathbf{y} = \sum_i \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})^\top \beta$, where $\beta = [\beta_1, ...\beta_d]^\top \in \Re^d$, $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), ...h_d(\mathbf{x})]$ is a set of basis functions. $h_i(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_i)$, where $K(\cdot)$ is a certain symmetric kernel function.

Typically, it is assumed that the output variable $\mathbf{y}$ from the training set was contaminated by additive white Gaussian noise, *i.e.* $\mathbf{y}_i = f(\mathbf{x}_i, \beta) + w_i$, for $i = 1, ..., n$, where $[w_1, ..., w_n]$ is a set of i.i.d. white Gaussian random variable with variance σ^2. Thus, the conditional probability $p(\mathbf{y}|\beta)$ is Gaussian, *i.e.* $p(\mathbf{y}|\beta) = \mathbf{N}(\mathbf{y}|\mathbf{h}(\mathbf{x})^\top \beta, \sigma^2 \mathbf{I})$. We write $\mathbf{H} = \mathbf{h}(\mathbf{x})^\top$, where $\mathbf{H}$ is called design matrix.

Simply apply Maximum Likelihood Estimator (MLE), we get least square error estimation, $\hat{\beta} = (\mathbf{H}^\top \mathbf{H})^{-1} \mathbf{H}^\top \mathbf{y}$. Note there is not any preference on β, so its prior is a uniform distribution. With a zero-mean Gaussian prior for β with variance A, the estimation is turned into *maximum a posteriori* (MAP) process. The prior of β then becomes an ℓ_2-norm regularization term in the log-likelihood, where it will prefer a small β. When $A = \mu^2 I$, it is called *ridge regression* [1].

Other β prior can also be applied. If sparseness is preferred, then Laplacian prior can be adopted for β, *i.e.*

$$p(\beta|\alpha) = \left(\frac{\alpha}{2}\right)^k \exp(-\alpha \|\beta\|_1),$$

where α is a parameter of the Laplacian pdf, and $\|\mathbf{x}\|_i$, $i = 0, ...\infty$ is the so called ℓ_i-norm. Laplacian distribution features heavy tail and has a high concentration at near-zero area, which means that most of the β's components will be zero, and the probability of having a large value is relatively high, comparing to the Gaussian distribution with the same variance. Utilizing the same MAP process, the estimation of β is given by

$$\hat{\beta} = \arg\min_\beta \left\{ \|\mathbf{y} - \mathbf{H}\beta\|_2^2 + t \|\beta\|_1 \right\}, \qquad (1)$$

where $t = 2\sigma^2\alpha$ is a control parameter, which can favor either the squared error term or the ℓ_1-norm regularization term. This criterion is also known as LASSO [2]. It is worth noting here that due to the non-Gaussian prior, the MAP estimation is not equivalent to the Bayesian estimation as it is in the ridge regression. So the estimation is biased. To make a unbiased estimation, one needs to integrate in all β space, which is computationally prohibitive. However, if the posterior concentrates at certain point, then this biased estimation may only have a small variance from the unbiased estimation, which is desirable. This is one reason why a concentrated sparse prior is preferred. Some researchers introduced "hyper-parameter" to further steepen the prior, like in Relevance Vector Machine (RVM) [3] and Figueiredo's work [4].

Another reason that a sparse representation is desirable is because it improves the generalization of a learning system. For example in supervised learning, the goal is to infer a mapping based on the training samples. The generalization capability is accomplished by reducing the complexity of the model, which is characterized by the number of non-zero parameters. This problem is formalized as ℓ_0-norm

minimization. It is known that ℓ_0-norm minimization is NP-hard [5]. However, it is established that the solution of the ℓ_1 problem is the same as the ℓ_0 problem if certain condition is satisfied. So, ℓ_1-norm problem is still an important issue.

In this paper, we proposed a method to solve the ℓ_1-norm version of the problem. We construct the dual problem of LASSO criterion as in Eq. (1) and use a gradient-based method to find the solution. This method is by no means claimed to be superior to the quadratic programming (QP) based method, however it opens a perspective to address the problem differently. Its competitive learning nature can also motivate other biologically plausible models for solving similar problems.

The reminder of paper is organized as follows: In Section 2 we formulate the dual problem of the LASSO, and propose a solution base on gradient descend. We reformulate the proposed algorithm in Section 2.3. The experiment results and comparison with existing sparse optimization algorithms is in Section 3. Section 4 provides conclusions.

2. METHOD

2.1. Duality

First, we will construct the dual problem of Eq. (1). Fig. 1 illustrates this problem. The Eq. (1) can be geometrically explained as the minimum ℓ_1-norm between the origin and the convex set K. This distance, according to duality theory, is equal to the maximum ℓ_∞-norm distance between the origin and the plane that separate the origin and the convex set K.

Theorem 2.1 *If we have*

$$\hat{\beta}' = \arg\max_{\beta} - \frac{\begin{bmatrix} \partial Z/\partial t\beta \\ -1 \end{bmatrix}^T \begin{bmatrix} t\beta \\ Z \end{bmatrix}}{\left\| \begin{bmatrix} \partial Z/\partial t\beta \\ -1 \end{bmatrix} \right\|_\infty},$$

where $Z = \|\mathbf{y} - \mathbf{H}\beta\|_2^2$, *then* $\hat{\beta}' = \hat{\beta}$, *and* $\hat{\beta}$ *is the same as in Eq.* (1).

The generalized proof can be found in [6].

2.2. Derivations

Let $Z = \|\mathbf{y} - \mathbf{H}\beta\|_2^2 = \beta^\top \mathbf{H}^\top \mathbf{H}\beta - 2\mathbf{y}^\top \mathbf{H}\beta + \mathbf{y}^\top \mathbf{y}$, and

$$J(\beta) = \begin{bmatrix} \frac{\partial Z}{\partial t\beta} \\ -1 \end{bmatrix}^\top = \begin{bmatrix} 2/t(\mathbf{H}^\top \mathbf{H}\beta - \mathbf{H}^\top \mathbf{y}) \\ -1 \end{bmatrix}^\top \quad (2)$$

Now we can formulate the dual problem of the original LASSO criterion,

$$E(\beta) = -\frac{J(\beta) \begin{bmatrix} t\beta \\ Z \end{bmatrix}}{\|J(\beta)\|_\infty}, \quad (3)$$

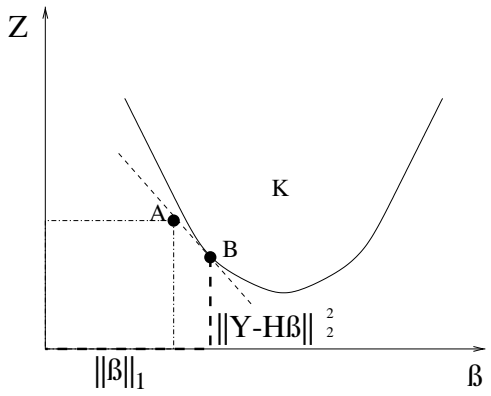

Fig. 1. Geometry explanation of duality. Point B is the closest point in K to origin.

Therefore,

$$\frac{\partial Z}{\partial t\beta} = \frac{2}{t}(\mathbf{H}^\top \mathbf{H}\beta - \mathbf{H}^\top \mathbf{y}). \quad (4)$$

$$\frac{\partial J}{\partial \beta} = \begin{bmatrix} & 0 \\ \frac{2}{t}\mathbf{H}^\top \mathbf{H} & \vdots \\ & 0 \end{bmatrix}. \quad (5)$$

Letting $\mathbf{C} = \mathbf{H}^\top \mathbf{H} = [\mathbf{c}_1, ..., \mathbf{c}_n]$, and plugging with Eq. (5) and Eq. (4), we get

$$\frac{\partial E}{\partial \beta} = \frac{2\mathbf{C}\beta}{\|J(\beta)\|_\infty} - \frac{\left[\beta^\top \mathbf{C}\beta - \mathbf{y}^\top \mathbf{y}\right]}{\|J(\beta)\|_\infty^2} \cdot \frac{\partial \|J(\beta)\|_\infty}{\partial \beta}.$$

Now we proceed to compute the partial derivative of $\|J(\beta)\|_\infty$ w.r.t. β.

$$\|J(\beta)\|_\infty = \max_i \left\{ \left| \frac{2}{t} \left(\beta^\top \mathbf{c}_i - \mathbf{y}^\top \mathbf{h}_i\right) \right|, 1 \right\}$$
$$= \sqrt{\max_i \left\{ \left| \frac{2}{t} \left(\beta^\top \mathbf{c}_i - \mathbf{y}^\top \mathbf{h}_i\right) \right|^2, 1 \right\}}$$

So,

$$\frac{\partial \|J(\beta)\|_\infty}{\partial \beta} = \frac{1}{2} \|J(\beta)\|_\infty^{-1} \cdot \frac{\partial}{\partial \beta} \max_i \left\{ \left| \frac{2}{t} \left(\beta^\top \mathbf{c}_i - \mathbf{y}^\top \mathbf{h}_i\right) \right|^2, 1 \right\}. \quad (6)$$

The last partial derivative term in Eq. (6) needs some special treatment. The difficulty of the analysis lies in the discontinuity caused by the maximum function. This problem can be circumvented by the use of the following equality. Let $\{a_i\}$ be a set of positive real scalars; then it generally holds that

$$\max_i \{a_i\} \equiv \lim_{r \to \infty} \left[\sum_i a_i^r \right]^{\frac{1}{r}}.$$

This is just another identity of the ℓ_∞-norm, which is differentiable. We have not yet get the strict derivation of this method. In fact, a similar technique can be seen in [7], in which Kohonen derived the vector quantization (VQ) algorithm base on this idea. In [8], a competitive learning algorithm has been derived from a maximum criteria function. Based on aforementioned observation, we conjecture that the order of the partial derivative and the max function can be exchanged. This leads to the following updating rules.

$$\frac{\partial}{\partial\beta}\max_i\left\{\left|\frac{2}{t}\left(\beta^\top\mathbf{c}_i-\mathbf{y}^\top\mathbf{h}_i\right)\right|^2,1\right\}=$$
$$\begin{cases}\frac{8}{t}\left(\beta^\top\mathbf{c}_m-\mathbf{y}^\top\mathbf{h}_m\right)\cdot\mathbf{c}_m, \text{if } \frac{4}{t^2}\left(\beta^\top\mathbf{c}_m-\mathbf{y}^\top\mathbf{h}_m\right)^2>1\\ [0,0...0]^\top, \text{otherwise.}\end{cases}, \tag{7}$$

where $m=\arg\max_j\left(\left|\beta^\top\mathbf{c}_j-\mathbf{y}^\top\mathbf{h}_j\right|\right)$.

Rearrange these equations, we have the final updating rules,

$$\nabla\beta\propto\begin{cases}\frac{2\mathbf{C}\beta}{\|J(\beta)\|_\infty}-\frac{4\left[\beta^\top\mathbf{C}\beta-\mathbf{y}^\top\mathbf{y}\right]}{t\|J(\beta)\|_\infty^3}\cdot\left(\beta^\top\mathbf{c}_m-\mathbf{y}^\top\mathbf{h}_m\right)\mathbf{c}_m\\ \frac{2\mathbf{C}\beta}{\|J(\beta)\|_\infty}\end{cases} \tag{8}$$

The switching condition of these two updating rules is the same as that in Eq. (7)

2.3. Gradient Sparseness Optimization Algorithm

We summarize the training procedure in Algorithm 1. Each iteration of this algorithm is computational efficient, because it only involves matrix multiplication and maximum function. The time complexity of each iteration is $O(kn)$, where k is number of basis vector and m is the dimension of each basis vector.

Algorithm 1 Gradient Sparseness Optimization Algorithm

1: Preprocessing: Get the training set $(\mathbf{x}_i,\mathbf{y}_i)$, $i=0,1,2,...,n$. For each i, $\mathbf{y}_i=\mathbf{y}_i-\bar{\mathbf{y}}$, where $\bar{\mathbf{y}}=\frac{1}{m}\sum_{i=1}^m\mathbf{y}_i$.
2: $\beta_i=0, i=0,1,2,...,k$.
3: Initialize the design matrix $\mathbf{H}$, and compute $\mathbf{C}=\mathbf{H}^\top\mathbf{H}$.
4: **repeat**
5: Compute $m=\arg\max_j\left(\left|\beta^\top\mathbf{c}_j-\mathbf{y}^\top\mathbf{h}_j\right|\right)$.
6: **if** $\frac{4}{t^2}\left(\beta^\top\mathbf{c}_m-\mathbf{y}^\top\mathbf{h}_m\right)^2>1$ **then**
7: Update β with:
$$\nabla\beta\propto\frac{2\mathbf{C}\beta}{\|J(\beta)\|_\infty}-\frac{4\left[\beta^\top\mathbf{C}\beta-\mathbf{y}^\top\mathbf{y}\right]}{t\|J(\beta)\|_\infty^3}\cdot\left(\beta^\top\mathbf{c}_m-\mathbf{y}^\top\mathbf{h}_m\right)\mathbf{c}_m$$
8: **else**
9: Update β with:
$$\nabla\beta\propto\frac{2\mathbf{C}\beta}{\|J(\beta)\|_\infty}$$
10: **end if**
11: **until** Objective function in Eq. (3) reaches the target value.

3. EXPERIMENTS

3.1. Kernel Regression

Our first experiment illustrates the performance of the proposed algorithm in kernel regression. The regression model is

$$\mathbf{y}=f(\mathbf{x},\beta)=\beta_0+\sum_{i=1}^k\beta_iK(\mathbf{x},\mathbf{x}_i),$$

where $K(\mathbf{x},\mathbf{x}_i)=exp\{-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\sigma^2}\}$ is the kernel function, $\mathbf{x}_i$ and σ are the kernel parameters. The function to be approximated is $1-d$ sinc function $y=\sin(x)/x$. We randomly collected 150 samples and added Gaussian noise with variance 0.01. The first row in Fig. 2 shows the fitting results of proposed method, ridge regression and LASSO regression, respectively. The dots are samples with noise, and the dashed lines are the ground truth sinc function. Solid lines show the approximation results. The circled dots correspond to the kernels with nonzero weight, a.k.a the "supporting kernels". In the second row, we use the bar figure to display the weights of those kernels. As it clearly indicated, both proposed method and LASSO achieve sparseness. The ℓ_∞-norms are also marked on these figures. The proposed method in our testing performs better than LASSO regression.

Fig. 3 illustrate how the control parameter $t=2\sigma^2\alpha$ affects the mean square error and model sparseness. We conducted 20 tests with t ranging from 0.3 to 1.5. As indicated in the figure, greater t makes the model fit well but increases its complexity, and vice versa.

3.2. Classification

The experiment addressed the kernel-based classifier for two-class problems: A special case of the regression problem with $y\in\{+,-\}$. The classifier is formulated as the following two functions:

$$p(+\mid\mathbf{x})=\psi(\mathbf{H}\beta_+)$$
$$p(-\mid\mathbf{x})=\psi(\mathbf{H}\beta_-)$$

where ψ denotes the logistic function. If $p(+\mid\mathbf{x})>p(-\mid\mathbf{x})$, $\mathbf{x}$ belongs to the class $+$. Otherwise, $\mathbf{x}$ belongs to the class $-$.

We used two data sets from real-data problems: the *Pima indian diabetes*[1], which were collected from women of Pima

[1]Downloadable at www.stats.ox.ac.uk/pub/PRNN

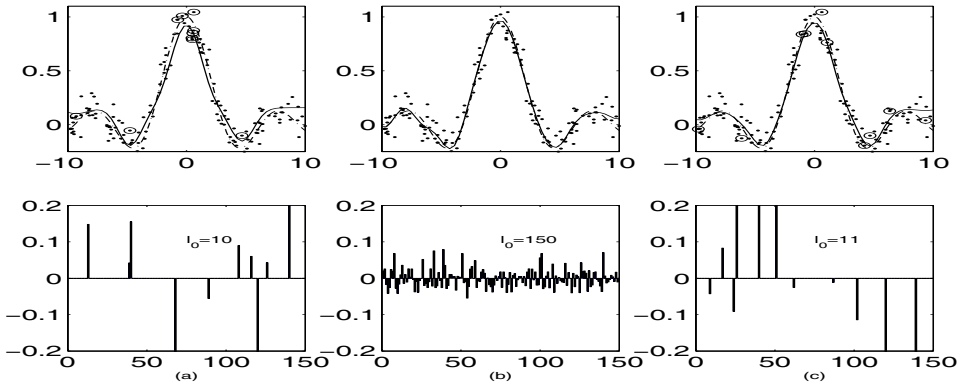

Fig. 2. Kernel regression results. The dashed lines are true sinc functions. Solid lines are approximation results. (a) Proposed gradient method. (b) Ridge regression. (c) LASSO regression.

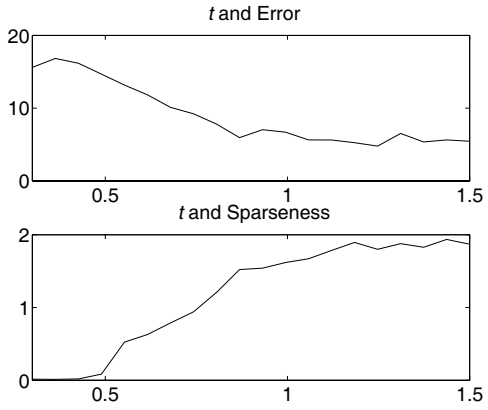

Fig. 3. The effect of control parameter t over mean square error and sparseness.

heritage and the goal is to decide whether a subject has diabetes or not, based on 7 different tests; the *Wisconsin breast cancer* (WBC)[2], whose goal is to diagnose (benigh/malignant) based on the results of 9 measurements. In WBC, we removed the cases with missing attributes for simplicity. Table 1 shows the results of the proposed classifier on the 5-fold cross validation experiment. For comparison, we also include the results of the kernel-based logistic classifier. In both data sets, the proposed classifier is better. The performance is improved partially by setting some decayed kernel weight to be exact zero.

4. CONCLUSIONS

In this paper, we have formulated the dual problem of the ℓ_1 norm based sparse approximation. We show the geometric relation of the duality and solve the dual ℓ_∞ maximization problem by gradient. The algorithm's performance is close to or better than the result of LASSO regression.

[2]Downloadable at www.ics.uci.edu/ mlearn/MLSummary.html

Table 1. The result of the 5-fold cross-validation.

ROC/No. kernels	Pima	WBC
Logistic	0.750/200	0.772/455
Proposed classifier	0.965/70	0.980/253

5. REFERENCES

[1] A. Hoerl and R. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, pp. 55–67, 1970.

[2] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal Statistical Society (B)*, vol. 58, pp. 267–288, 1996.

[3] M. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[4] M.Ã.T. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1150–1159, 2003.

[5] E. Amaldi and V. Kann, "On the approximibility of minimizing non zero variables or unsatisfied relations in linear systems," *Theoretical Computer Science*, vol. 209, pp. 237–260, 1998.

[6] D.G̃. Luenverger, *Optimization by Vector Space Methods*, John Wiley and Sons, Inc., 1969.

[7] T. Kohonen, *Self-organizing Maps*, pp. 60–62, Springer-Verlag, New York, 2001, 3rd edition.

[8] J. Weng and N. Zhang, "A quasi-optimally efficient algorithm for independent component analysis," in *Proc. IEEE Int. Conf. on Acoustics Speech, and Signal Processing*, Montreal, Canada, 2004.