An Efficient Algorithm For Weak Hierarchical Lasso

Yashu Liu
Arizona State University
Tempe, AZ 85287
Yashu.Liu@asu.edu

Jie Wang
Arizona State University
Tempe, AZ 85287
jie.wang.ustc@asu.edu

Jieping Ye
Arizona State University
Tempe, AZ 85287
Jieping.Ye@asu.edu

ABSTRACT

Linear regression is a widely used tool in data mining and machine learning. In many applications, fitting a regression model with only linear effects may not be sufficient for predictive or explanatory purposes. One strategy which has recently received increasing attention in statistics is to include feature interactions to capture the nonlinearity in the regression model. Such model has been applied successfully in many biomedical applications. One major challenge in the use of such model is that the data dimensionality is significantly higher than the original data, resulting in the small sample size large dimension problem. Recently, weak hierarchical Lasso, a sparse interaction regression model, is proposed that produces sparse and hierarchical structured estimator by exploiting the Lasso penalty and a set of hierarchical constraints. However, the hierarchical constraints make it a non-convex problem and the existing method finds the solution of its convex relaxation, which needs additional conditions to guarantee the hierarchical structure. In this paper, we propose to directly solve the non-convex weak hierarchical Lasso by making use of the GIST (General Iterative Shrinkage and Thresholding) optimization framework which has been shown to be efficient for solving non-convex sparse formulations. The key step in GIST is to compute a sequence of proximal operators. One of our key technical contributions is to show that the proximal operator associated with the non-convex weak hierarchical Lasso admits a closed form solution. However, a naive approach for solving each subproblem of the proximal operator leads to a quadratic time complexity, which is not desirable for large-size problems. To this end, we further develop an efficient algorithm for computing the subproblems with a linearithmic time complexity. We have conducted extensive experiments on both synthetic and real data sets. Results show that our proposed algorithm is much more efficient and effective than its convex relaxation.

KDD'14, August 24–27, 2014, New York, NY, USA.
Copyright 2014 ACM 978-1-4503-2956-9/14/08 ...$15.00.
http://dx.doi.org/10.1145/2623330.2623665.

Categories and Subject Descriptors

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

General Terms

Algorithms

Keywords

Sparse learning; non-convex; weak hierarchical Lasso; proximal operator

1. INTRODUCTION

Consider a linear regression model with the outcome variable y and d predictors $x_1, \ldots, x_d$:

$$y = w_0 + \sum_{i=1}^{d} x_i w_i + \epsilon, \qquad (1)$$

where w_0 is the bias term, $w_i, i = 1 \ldots, d$ is the coefficient and $\epsilon \sim N(0, \sigma^2)$ is the noise term. In many applications, a simple linear regression model is not sufficient for predictive or explanatory purposes. One strategy which has recently received increasing attention in statistics is to include interaction terms into the model to capture the nonlinearity of the data [17, 22]. For example, the linear model including terms of order-2 and lower has the following form:

$$y = w_0 + \sum_{i=1}^{d} x_i w_i + \frac{1}{2} \sum_{i=1}^{d} \sum_{j=1}^{d} x_i x_j Q_{i,j} + \epsilon, \qquad (2)$$

where the cross-product term $x_i x_j$, $i \neq j$ refers to as the interaction variable (one may view x_i^2 as a special interaction variable), and $w_i's$ and $Q \in \mathbb{R}^{d \times d}$ are the main effect and interaction effect coefficients respectively. Applications with interaction regression models are omnipresent. For example, in psychological study, the effectiveness of using 3-way interactions was demonstrated in testing psychological hypothesis [9]; there are strong evidences found in [4] that genetic-environment interactions have significant effects on conduct disorders; the research in [11] found a couple of evidences of gene-environment interactions in predicting depression status; in [26], the interaction between continuance commitment and affective commitment was found significant in predicting job withdraw intentions and absenteeism; [13] discovered that brain-derived neurotrophic factor interacts with early life stress in predicting cognitive features of depression and anxiety.

However, the use of higher order terms leads to data of high dimensionality. For instance, for regression model (1), if one wants to add all terms of order-k and lower, then there will be a total of $\mathcal{O}(d^k)$ variables, which is computationally demanding for parameter estimation even when k and d are fairly small. Thus, an efficient approach that is able to deal with huge dimensionality is desired in such cases, and the sparse learning methodology is one promising approach for tackling such problem [27, 18, 7, 5, 32]. In this paper, we focus on the model (2) with pairwise interactions, *i.e.,* two-factor interactions. Note that the analysis can be extended to the model with higher-order interactions.

In general, not all of the main effects and interactions are of interest, thus it is critical to select the variables of great significance. One simple approach for high dimensional interaction regression is to directly apply the Lasso [27], also known as the "all-pairs Lasso" [2] in the case of two-factor interactions. However, the all-pairs Lasso estimator does not account for any structural information which has been shown to be important for prediction and interpretation of the high dimensional interaction regression model [2, 30, 25, 29, 6]. In statistics, a hierarchical structure between main effects and interaction effects has been shown to be very effective in constraining the search space and identifying important individual features and interactions [2, 30, 25, 29, 6]. Specifically, the hierarchical constraint requires that an interaction term $x_i x_j$ is selected in the model only if the main effects x_i and/or x_j are included. Strong theoretical properties have been established for such hierarchical model [29, 30]. The hierarchical structure is supported by the argument that large main effects may result in interaction of more importance, and it is desired in a wide range of applications in engineering and underlying science. Traditional approaches to fit such a model typically follow the following two-step procedures [22]:

(i) Fit a linear regression model that only includes the main effects and then select the significant features;

(ii) Fit the reformulated model with the identified individual features and the interactions constructed via domain knowledge.

Since even a small d may lead to a huge amount of interaction variables, the two-step procedure is still time-consuming in many applications. Recently, there have been growing research efforts on imposing the hierarchical structure on main effects and interactions in the regression model with novel sparse learning methods. In [2], in order to enable feature selection and impose heredity structures, the authors proposed strong hierarchical Lasso which adds a set of constraints to the Lasso formulation to achieve the *strong hierarchy* where the interaction effects are non-zero only if the corresponding main effects are non-zero. In [25], a Lasso-type penalized least square formulation called VANISH was proposed to achieve the strong hierarchy between the interaction effects and main effects. In [29], a type of non-negative garrote method was proposed to achieve the heredity structures. In [30], the Composite Absolute Penalties were proposed to achieve heredity structures for interaction models. In contrast to the above works which fulfill the hierarchical structure via solving convex problems, Choi *et al.* in [6] formulated a non-convex problem to achieve the strong hierarchy by assuming that the coefficient of an interaction term is a product of a scalar and main effect coefficients. Different from the strong hierarchy, the *weak hierarchy* between the main effects and the interaction effects requires that an interaction is included in the model only if at least one of the main effects is included in the model. In mathematical form, $Q_{i,j} \neq 0$ only if $w_i \neq 0$ OR $w_j \neq 0$. The weak hierarchy can be considered as a structure in between the strong hierarchy and no hierarchical structure [2, 29, 30]. Specifically, weak hierarchy allows those interactions with only one significant "parent" (main effect) to be included in the model. Several existing empirical studies have demonstrated the stronger predictive power of weak hierarchical model [19]. In our study, we mainly focus on the interaction regression model with weak hierarchical structure.

We follow the weak hierarchical Lasso approach recently proposed by [2] to fit the pairwise interaction regression model with the weak hierarchy. By imposing restrictions of the weak hierarchy and taking advantage of the Lasso penalty [27] that leads to sparse coefficients, the weak hierarchical Lasso is able to simultaneously attain a hierarchical solution and identify important main effects and interactions. However, the set of constraints restricting hierarchical constraints make the problem non-convex; the algorithm proposed in [2] aims to solve a convex relaxation. The convex relaxation, however, requires additional conditions to guarantee the weak hierarchy, which is not desirable.

In this paper, we propose to directly solve the weak hierarchical Lasso using the GIST (General Iterative Shrinkage and Thresholding) optimization framework recently proposed by [15]. The GIST framework has been shown to be highly efficient for solving large-scale non-convex problems. The most critical step in GIST is to compute a sequence of proximal operators [23]. In this paper, we first show that the proximal operator related to weak hierarchical Lasso admits an analytical form solution by factorizing unknown coefficients into sign matrices and non-negative coefficients. However, a naive method of computing the subproblem of the proximal operator leads to a quadratic time complexity, which is not desirable for large-size problems. To this end, we further develop an efficient algorithm for solving the subproblems, which achieves a linearithmic time complexity. We evaluate the efficiency and effectiveness of the proposed algorithm and compare it with the convex relaxation in [2] and other state-of-the-art methods using synthetic and real data sets. Our empirical study demonstrates the high efficiency of our algorithm and the superior predictive performance of weak hierarchical Lasso over the competing methods.

The remaining of the paper is organized as follows: we give a brief review of the weak hierarchical Lasso and its convex relaxation in Section 2. In Section 3, we derive the closed form solution to the proximal operator of the original weak hierarchical Lasso by decomposing the unknown coefficients into signs and the non-negative coefficients. Then, we show how the associated proximal operator can be computed efficiently. We report the experimental results in Section 4. We conclude this paper in Section 5.

2. THE WEAK HIERARCHICAL LASSO

In this section, we briefly review the weak hierarchical Lasso and its corresponding convex relaxed formulation [2]. Suppose we are given n pairs of data points $\{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^d \times \mathbb{R}$. Let $Y \in \mathbb{R}^{n\times 1}$ be the vector of outcome and $X \in$

$\mathbb{R}^{n\times d}$ be the design matrix. Let $Z \in \mathbb{R}^{n\times(d\cdot d)}$ be the matrix of interactions where

$$Z = \left[Z^{(1)}, Z^{(2)}, \ldots, Z^{(d)}\right],$$

$Z^{(i)} \in \mathbb{R}^{n\times d}$ and each column of $Z^{(i)}, i = 1, \ldots d$ is an interaction, *i.e.,* $Z^{(i)}_{\cdot,j} = X_{\cdot,i} \odot X_{\cdot,j}$ ($\odot$ is the operator of element-wise product). Thus, $Z^{(i)}$ captures the pairwise interactions between the i-th feature and all d features. Note that, we include the quadratic terms x_i^2 in the interaction model for clearer presentation, however our analysis is still applicable if they are not included in the model. By assuming that Y is centered and X, Z are column-wise normalized to zero mean and unit standard deviation, we can set the bias term $w_0 = 0$. Thus, in matrix form, the pairwise interaction regression model can be expressed as

$$Y = Xw + \frac{1}{2} Z \cdot \mathrm{vec}(Q) + \epsilon, \tag{3}$$

where $\epsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I})$ and "vec" is the vectorization operator that transforms a matrix to a column vector by stacking the columns of the matrix. Thus, the least square loss function of (3) is given by:

$$\mathcal{L}(w, Q) = \frac{1}{2}\left\| Y - Xw - \frac{1}{2} Z \cdot \mathrm{vec}(Q) \right\|_2^2. \tag{4}$$

Then, the weak hierarchical Lasso formulation takes the form of [2]:

$$\begin{aligned} &\min_{x,Q}\ \mathcal{L}(w, Q) + \lambda\|w\|_1 + \frac{\lambda}{2}\|Q\|_1 \\ &\text{s.t.}\quad \|Q_{\cdot,j}\|_1 \le |w_j| \quad \text{for} \quad j = 1, \ldots, d, \end{aligned} \tag{5}$$

where $\|Q\|_1 = \sum_{i,j} |Q_{i,j}|$ and λ is the Lasso penalty parameter.

Note that the constraints in (5) guarantee the weak hierarchical structure since the coefficient $Q_{i,j}$ of interaction $x_i x_j$ is non-zero only if at least one of its main effects is included in the model, *i.e.,* $w_i \ne 0$ or $w_j \ne 0$. However, the imposed hierarchical constraints make problem (5) non-convex. Instead of solving (5), Bien *et al.* in [2] proposed to solve the following relaxed version:

$$\begin{aligned} &\min_{w,Q}\ \mathcal{L}(w^+ - w^-, Q) + \lambda\mathbf{1}^T(w^+ + w^-) + \frac{\lambda}{2}\|Q\|_1 \\ &\left.\begin{aligned} \text{s.t.}\quad &\|Q_{\cdot,j}\|_1 \le w_j^+ + w_j^- \\ &w_j^+ \ge 0 \\ &w_j^- \ge 0 \end{aligned}\right\} \quad \text{for} \quad j = 1, \ldots, d, \end{aligned} \tag{6}$$

where $\mathbf{1}$ represents a column vector of all ones. In view of (6), we can see that $\|w\|_1$ is relaxed to $w^+ + w^-$. Problem (6) is convex and can be solved by many efficient solvers such as FISTA [1]. However, Bien *et al.* in [2] showed that problem (6) needs an additional ridge penalty to guarantee the weak hierarchical structure of the estimator. In this paper, we propose an efficient algorithm which directly solves the non-convex weak hierarchical Lasso formulation in (5).

3. THE PROPOSED ALGORITHM

In this section, we propose an efficient algorithm named "eWHL", which stands for "**e**fficient **W**eak **H**ierarchical **L**asso", to directly solve the weak hierarchical Lasso. eWHL makes use of the optimization framework of GIST (General Iterative Shrinkage and Thresholding) due to its high efficiency and effectiveness for solving non-convex sparse formulations. One of the critical steps in GIST is to compute the proximal operator associated with the penalty functions. As one of our major contributions, we first factorize the unknown coefficients into the product of their signs and magnitudes; and then show that the proximal operator of (5) admits a closed form solution in Section 3.1. Another major contribution is that we present an efficient algorithm for computing the proximal operator associated with the non-convex weak hierarchical Lasso in Section 3.2. The time complexity of solving each subproblem of the proximal operator can be reduced from quadratic to linearithmic. We then summarize our algorithm for computing the proximal operator in Section 3.2.

3.1 The Closed Form Solution to the Proximal Operator

In this section, we show how to derive the closed form solution of the proximal operator associated with (5) in detail. Let

$$\mathcal{P} = \left\{(a, B),\ a \in \mathbb{R}^d,\ B \in \mathbb{R}^{d\times d}\ \middle|\ \|B_{\cdot,j}\|_1 \le |a_j|,\ j = 1, \ldots, d\right\}$$

and the indicator function be defined by

$$\mathcal{R}(w, Q) = \begin{cases} \lambda\|w\|_1 + \frac{\lambda}{2}\|Q\|_1, & \text{if} \quad (w, Q) \in \mathcal{P} \\ +\infty, & \text{if} \quad (w, Q) \notin \mathcal{P} \end{cases}. \tag{7}$$

Given a sequence $\left\{\left(w^{(k)}, Q^{(k)}\right)\right\}$, the proximal operator associated with weak hierarchical Lasso is:

$$\begin{aligned} &\left(w^{(k+1)}, Q^{(k+1)}\right) \\ &= \arg\min_{w,\ Q}\ \mathcal{L}\left(w^{(k)}, Q^{(k)}\right) + \left\langle \nabla_w \mathcal{L}\left(w^{(k)}, Q^{(k)}\right),\ w - w^{(k)} \right\rangle \\ &\quad + \left\langle \nabla_Q \mathcal{L}\left(w^{(k)}, Q^{(k)}\right),\ Q - Q^{(k)} \right\rangle + \frac{t^{(k)}}{2}\left\| w - w^{(k)} \right\|_2^2 \\ &\quad + \frac{t^{(k)}}{2}\left\| Q - Q^{(k)} \right\|_F^2 + \mathcal{R}(w, Q), \end{aligned} \tag{8}$$

where $t^{(k)} > 0$.

Simple algebraic manipulation leads to

$$\begin{aligned} \left(w^{(k+1)}, Q^{(k+1)}\right) = \arg\min_{w,\ Q}\ &\frac{1}{2}\left\| w - v^{(k)} \right\|_2^2 + \frac{1}{2}\left\| Q - U^{(k)} \right\|_2^2 \\ &+ \frac{1}{t^{(k)}}\mathcal{R}(w, Q), \end{aligned} \tag{9}$$

where

$$\begin{aligned} v^{(k)} &= w^{(k)} - \nabla_w \mathcal{L}\left(w^{(k)}, Q^{(k)}\right)/t^{(k)}, \\ U^{(k)} &= U^{(k)} - \nabla_Q \mathcal{L}\left(w^{(k)}, Q^{(k)}\right)/t^{(k)}. \end{aligned}$$

Thus, problem (5) can be solved by iteratively solving the proximal operator in (9). Because $\mathcal{R}(w, Q)$ is an indicator function, we can rewrite the proximal operator (9) as

$$\begin{aligned}&\arg\min_{w,\ Q}\ \frac{1}{2}\|w-v\|_2^2+\frac{1}{2}\|Q-U\|_F^2+\frac{\lambda}{t}\|w\|_1+\frac{\lambda}{2t}\|Q\|_1\\ &\qquad \text{s.t. } \|Q_{\cdot,j}\|_1\le |w_j| \quad \text{for} \quad j=1,\dots,d.\end{aligned} \tag{10}$$

We omit the superscripts for notational simplicity.

The vector of main effect coefficients can be written as

$$w=S^0\tilde{w},$$

where $\tilde{w}_j=|w_j|, j=1,\dots,d$ and $S^0\in\mathbb{R}^{d\times d}$ is a diagonal matrix whose j-th diagonal element is the sign of w_j, *i.e.,* $S^0_{j,j}=\text{sign}(w_j)$. We define

$$\text{sign}(w)=\begin{cases}1 & \text{if} \quad w>0\\ -1 & \text{if} \quad w<0\\ 0 & \text{if} \quad w=0\end{cases}, \tag{11}$$

and we assume in this paper that the sign operator is applied on vectors or matrices elementwise. Similarly, we factorize each column of the interaction coefficient matrix as $Q_{\cdot,j}=S^j\widetilde{Q}_{\cdot,j},\ j=1\dots,d$, where $\widetilde{Q}_{i,j}=|Q_{i,j}|$ and $S^j\in\mathbb{R}^{d\times d}$ is the diagonal sign matrix. Then, the proximal operator (10) is equivalent to

$$\begin{aligned}&\arg\min_{w,\ Q}\ \frac{1}{2}\|w-v\|_2^2+\frac{1}{2}\|Q-U\|_F^2+\frac{\lambda}{t}\|w\|_1+\frac{\lambda}{2t}\|Q\|_1\\ &\text{s.t.}\ \left.\begin{aligned}&\|Q_{\cdot,j}\|_1\le|w_j|\\ &w_j=S^0_{j,j}\tilde{w}_j\\ &Q_{\cdot,j}=S^j\widetilde{Q}_{\cdot,j}\\ &\tilde{w}_j\ge 0\\ &\widetilde{Q}_{\cdot,j}\succeq\mathbf{0}\end{aligned}\right\} \quad \text{for} \quad j=1,\dots,d,\end{aligned} \tag{12}$$

where $\widetilde{Q}$, $\tilde{w}$ and $S^j,\ j=0,\dots,d$ are the unknown variables, $\succeq$ is defined as the element-wise "greater than or equal to" comparison operator, *i.e.,* for $V,\ U\in\mathbb{R}^{d\times 1}, V\succeq U \Leftrightarrow V_i\ge U_i, i=1\dots,d$. Therefore, the solutions of the original weak hierarchical Lasso can be obtained by iteratively solving (12). Note that the amounts of l_1 penalties on w and Q can be different. Here we use the same penalty parameter λ for notational simplicity and consistency with the original formulation of weak hierarchical Lasso (5) studied in [2]. Though the factorization introduces more variables and constraints, we show that the resulting proximal operator admits a closed form solution. More importantly, we show that each sub-problem of the proximal operator can be solved by the proposed eWHL algorithm in linearithmic time. Indeed, the factorization of w and Q into their signs and magnitudes is the first key to directly solve the original weak hierarchical Lasso.

It is clear that the proximal operator in (12) can be decoupled into d subproblems:

$$\begin{aligned}&\arg\min_{\tilde{w}_j,S^0_{jj},\widetilde{Q}_{\cdot,j},S^j}\ \frac{1}{2}\left\|S^0_{jj}\tilde{w}_j-v_j\right\|_2^2+\frac{1}{2}\left\|S^j\widetilde{Q}_{\cdot,j}-U_{\cdot,j}\right\|_2^2\\ &\qquad\qquad+\frac{\lambda}{t}\tilde{w}_j+\frac{\lambda}{2t}\mathbf{1}^T\widetilde{Q}_{\cdot,j}\\ &\qquad \text{s.t.}\quad \left.\begin{aligned}\mathbf{1}^T\widetilde{Q}_{\cdot,j}\le\tilde{w}_j\\ \widetilde{Q}_{\cdot,j}\succeq\mathbf{0}\end{aligned}\right\},\quad \text{for } j=1,\dots,d.\end{aligned} \tag{13}$$

Next, we show that (13) has a closed form solution. Since

$$\begin{aligned}&\frac{1}{2}(w_j-v_j)^2=\frac{1}{2}\left(S^0_{jj}\tilde{w}_j-v_j\right)^2\\ =&\frac{1}{2}\left(S^0_{jj}\left(S^0_{jj}\tilde{w}_j-v_j\right)\right)^2=\frac{1}{2}\left(\tilde{w}_j-S^0_{jj}v_j\right)^2\end{aligned}$$

and $\tilde{w}_j\ge 0$, $S^0_{j,j}$ must have the same sign as v_j, that is, w_j has the same sign as v_j. Otherwise, the value of $\frac{1}{2}\left(\tilde{w}_j-S^0_{jj}v_j\right)^2$ will not achieve the minimum. Similarly, one can show that $S^j_{i,i}$, *i.e.,* the sign of $Q_{i,j}$, must be the same as the sign of $U_{i,j}$. Thus, the diagonal elements diag(S^0) = sign(v), diag(S^j) = sign($U_{\cdot,j}$), $j=1,\dots,d$. Next, we show how to compute $\tilde{w}$ and $\widetilde{Q}$.

By letting $\tilde{v}_j=S^0_{jj}v_j$ and $\widetilde{U}_j=S^jU_{\cdot,j}$, each subproblem (13) is equivalent to

$$\begin{aligned}&\arg\min_{\tilde{w}_j,\widetilde{Q}_{\cdot,j}}\ \frac{1}{2}\|\tilde{w}_j-\tilde{v}_j\|_2^2+\frac{1}{2}\left\|\widetilde{Q}_{\cdot,j}-\widetilde{U}_{\cdot,j}\right\|_2^2+\frac{\lambda}{t}\tilde{w}_j+\frac{\lambda}{2t}\mathbf{1}^T\widetilde{Q}_{\cdot,j}\\ &\qquad \text{s.t.}\quad \begin{aligned}&\mathbf{1}^T\widetilde{Q}_{\cdot,j}\le\tilde{w}_j\\ &\widetilde{Q}_{\cdot,j}\succeq\mathbf{0}\end{aligned}.\end{aligned} \tag{14}$$

After rearrangement, problem (14) can be expressed as:

$$\begin{aligned}&\min_{\tilde{w}_j,\widetilde{Q}_{\cdot,j}}\ \frac{1}{2}\|\tilde{w}_j-\check{v}_j\|_2^2+\frac{1}{2}\left\|\widetilde{Q}_{\cdot,j}-\check{U}_{\cdot,j}\right\|_2^2\\ &\qquad \text{s.t.}\quad \begin{aligned}&\mathbf{1}^T\widetilde{Q}_{\cdot,j}\le\tilde{w}_j\\ &\widetilde{Q}_{\cdot,j}\succeq\mathbf{0}\end{aligned},\end{aligned} \tag{15}$$

where $\check{v}_j=\tilde{v}_j-\frac{\lambda}{t}\mathbf{1}$ and $\check{U}_{\cdot,j}=\widetilde{U}_{\cdot,j}-\frac{\lambda}{2t}\mathbf{1}$.

We solve (15) by deriving its dual problem. Let $\gamma\ge 0$ be the Lagrangian multiplier dual variable of the first inequality constraint. Define the Lagrangian function of (15) as:

$$l(\gamma,\tilde{w},\widetilde{Q})=\frac{1}{2}(\tilde{w}-\check{v})^2+\frac{1}{2}\left\|\widetilde{Q}-\check{U}\right\|_2^2+\gamma\left(\mathbf{1}^T\widetilde{Q}-\tilde{w}\right)$$

where we omit the subscripts for simplicity with a little abuse of notation. Since the constraint $\mathbf{1}^T\widetilde{Q}\le\tilde{w}$ is affine, the strong duality holds for the minimization problem (15). Thus, the dual problem of (15) is:

$$\max_{\gamma\ge 0}\ \min_{\tilde{w},\widetilde{Q}\succeq\mathbf{0}}\ \frac{1}{2}(\tilde{w}-\check{v})^2+\frac{1}{2}\left\|\widetilde{Q}-\check{U}\right\|_2^2+\gamma\left(\mathbf{1}^T\widetilde{Q}-\tilde{w}\right). \tag{16}$$

By rearranging the terms, (16) is equivalent to:

$$\max_{\gamma\ge 0}\ \min_{\tilde{w},\widetilde{Q}\succeq\mathbf{0}}\ \frac{1}{2}(\tilde{w}-(\check{v}+\gamma))^2+\frac{1}{2}\left\|\widetilde{Q}-\left(\check{U}-\gamma\mathbf{1}\right)\right\|_2^2+h(\gamma), \tag{17}$$

where $h(\gamma)=-\check{v}\gamma-\frac{1}{2}\gamma^2+\gamma\mathbf{1}^T\check{U}-\frac{1}{2}\gamma^2\mathbf{1}^T\mathbf{1}$.

For fixed γ, in order to obtain the minimum of the objective function in (17), we conclude that

$$\begin{cases}\check{v}+\gamma\ge 0 & \Rightarrow \quad \tilde{w}=\check{v}+\gamma\\ \check{v}+\gamma<0 & \Rightarrow \quad \tilde{w}=0\\ \check{U}_i-\gamma\ge 0 & \Rightarrow \quad \widetilde{Q}_i=\check{U}_i-\gamma\\ \check{U}_i-\gamma<0 & \Rightarrow \quad \widetilde{Q}_i=0\end{cases} \tag{18}$$

due to the constraints $\tilde{w}\ge 0$ and $\widetilde{Q}\succeq\mathbf{0}$. Therefore, if we obtain a dual optimal solution γ^* that maximizes the dual problem (17), then we can readily compute the closed form solution to (13) and thus to (12). That is, $w^*=$

$S^0\tilde{w}^*$, $Q^*_{\cdot,j} = S^j\widetilde{Q}^*_{\cdot,j}$ where $\text{diag}(S^0) = \text{sign}(v_j)$, $\text{diag}(S^j) = \text{sign}(U_{\cdot,j})$, $j = 1, \dots, d$ and $\tilde{w}^*$, $\widetilde{Q}^*$ are obtained via (18) at the optimal dual solution γ^*.

3.2 The Dual Optimal Solution

Next, we show how to efficiently compute the dual optimal solution γ^*. First, we sort $-\breve{v}$ and $\breve{U}_i, i = 1, \dots, d$ in ascending order. Without loss of generality, we assume:

$$\breve{U}_1 \leq \ldots \leq \breve{U}_L \leq -\breve{v} \leq \breve{U}_{L+1} \leq \ldots \leq \breve{U}_d. \qquad (19)$$

There are four possible cases about the locations of γ. We discuss how to identify the optimal dual solution γ^* in each of the four cases.

Case 1 :
When $\ldots \leq \breve{U}_G \leq \gamma \leq \breve{U}_{G+1} \leq \ldots \leq -\breve{v} \leq \ldots$, the objective in (17) at γ^* becomes

$$\begin{aligned} &\frac{1}{2}\sum_{i=1}^{G}\left(\breve{U}_i - \gamma\right)^2 + \frac{1}{2}(\breve{v} + \gamma)^2 + h(\gamma) \\ =&\frac{1}{2}\sum_{i=1}^{G}\breve{U}_i^2 + \sum_{i=G+1}^{d}\gamma\breve{U}_i - \frac{d-G}{2}\gamma^2 + \frac{1}{2}\breve{v}^2. \end{aligned} \qquad (20)$$

Function (20) is a quadratic function with respect to γ and the unconstrained maximum is achieved at the axis of symmetry point $\frac{\sum_{i=G+1}^{d}\breve{U}_i}{d-G} \geq \breve{U}_{G+1}$. Since γ falls in the interval $\left[\breve{U}_G, \breve{U}_{G+1}\right]$, we set

$$\gamma = \breve{U}_{G+1}$$

to achieve the maximum objective value of (17). It can be further concluded that, in **Case 1**, among all the intervals on the left of $-\breve{v}$, the maximum objective value of (17) is achieved at the $\breve{U}_G$.

Case 2:
When $\ldots \leq \breve{U}_L \leq \gamma \leq -\breve{v} \leq \breve{U}_{L+1} \leq \ldots$, it turns out that the objective value in (17) at γ is similar to (20):

$$\frac{1}{2}\sum_{i=1}^{L}\breve{U}_i^2 + \sum_{i=L+1}^{d}\gamma\breve{U}_i - \frac{d-L}{2}\gamma^2 + \frac{1}{2}\breve{v}^2. \qquad (21)$$

By a similar argument, we can set $\gamma = -\breve{v}$ to achieve the maximum. Combining the results of **Case 1** and **Case 2**, we conclude that, we may only consider γ in the range $[\max\left(-\breve{v}, 0\right), +\infty]$. Note that when $L = d$, that is $\breve{U}_d \leq \gamma \leq -\breve{v}$, (21) is a constant $\frac{1}{2}\sum_{i=1}^{d}\breve{U}_i^2 + \frac{1}{2}\breve{v}^2$, and thus γ can be any value in the interval $\left[\breve{U}_d, -\breve{v}\right]$.

Case 3:
When $\ldots \leq \breve{U}_L \leq -\breve{v} \leq \gamma \leq \breve{U}_{L+1} \leq \ldots$, the value of the objective function in (17) at γ^* becomes

$$\begin{aligned} &\frac{1}{2}\sum_{i=1}^{L}\left(\breve{U}_i - \gamma\right)^2 + h(\gamma) \\ =&\frac{1}{2}\sum_{i=1}^{L}\breve{U}_i^2 + \gamma\left(\sum_{i=L+1}^{d}\breve{U}_i - \breve{v}\right) - \frac{d+1-L}{2}\gamma^2. \end{aligned} \qquad (22)$$

Again, (22) is a quadratic function of γ and $\frac{\sum_{i=L+1}^{d}\breve{U}_i - \breve{v}}{d+1-L} \geq -\breve{v}$. If $\frac{\sum_{i=L+1}^{d}\breve{U}_i - \breve{v}}{d+1-L} \geq \breve{U}_{L+1}$, the maximum is achieved at

$$\gamma = \breve{U}_{L+1},$$

otherwise the maximum is achieved at

$$\gamma = \frac{\sum_{i=L+1}^{d}\breve{U}_i - \breve{v}}{d+1-L}.$$

Case 4:
When $\ldots \leq -\breve{v} \leq \ldots \leq \breve{U}_G \leq \gamma \leq \breve{U}_{G+1} \leq \ldots$, the objective value in (17) is similar to (22):

$$\frac{1}{2}\sum_{i=1}^{G}\breve{U}_i^2 + \gamma\left(\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}\right) - \frac{p+1-G}{2}\gamma^2. \qquad (23)$$

If $\frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \geq \breve{U}_{G+1}$, then the maximum is achieved at

$$\gamma = \breve{U}_{G+1};$$

If $\frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \leq \breve{U}_G$, then the maximum is achieved at

$$\gamma = \breve{U}_G;$$

If $\breve{U}_G \leq \frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \leq \breve{U}_{G+1}$, the maximum is achieved at

$$\gamma = \frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G}.$$

Since we know exactly the value of γ for all the four cases, one naive way to find the optimal γ^* is to enumerate all the possible locations and pick the one that maximizes the objective function value in (17). However, evaluating the objectives for all possible locations from $\max(-\breve{v}, 0)$ to $\breve{U}_d$ leads to a quadratic time algorithm for solving (17). Interestingly, we show below that the time complexity of solving (17) can be reduced to $\mathcal{O}(d\log(d))$.

Let us first list some useful properties as follows:
Given the ordered sequence (19):

- **Property 1:**
The maximum objective value of (17) in Case 3 is larger than the one in Cases 1 & 2;
- **Property 2:**
In Case 4, for a pair of adjacent intervals $\left[\breve{U}_{G-1}, \breve{U}_G\right]$ and $\left[\breve{U}_G, \breve{U}_{G+1}\right]$, if $\frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \geq \breve{U}_{G+1}$ for $[\breve{U}_G, \breve{U}_{G+1}]$, then $\frac{\sum_{i=G}^{d}\breve{U}_i - \breve{v}}{d+1-(G-1)} \geq \breve{U}_G$ for $[\breve{U}_{G-1}, \breve{U}_G]$;
- **Property 3:**
In Case 4, if $\frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \geq \breve{U}_{G+1}$ for $\left[\breve{U}_G, \breve{U}_{G+1}\right]$, the maximum objective value of (17) in $\left[\breve{U}_G, \breve{U}_{G+1}\right]$ is larger than or equal to the one in $\left[\breve{U}_{G-1}, \breve{U}_G\right]$.
- **Property 4:**
In Case 4, for a pair of adjacent intervals $\left[\breve{U}_{G-1}, \breve{U}_G\right]$ and $\left[\breve{U}_G, \breve{U}_{G+1}\right]$, if we have $\frac{\sum_{i=G}^{d}\breve{U}_i - \breve{v}}{d+1-(G-1)} \leq \breve{U}_{G-1}$ for $\left[\breve{U}_{G-1}, \breve{U}_G\right]$, then $\frac{\sum_{i=G+1}^{d}\breve{U}_i - \breve{v}}{d+1-G} \leq \breve{U}_G$ for $\left[\breve{U}_G, \breve{U}_{G+1}\right]$.

- **Property 5:**
  In Case 4, if $\frac{\sum_{i=G}^{d} \check{U}_i - \check{v}}{d+1-(G-1)} \leq \check{U}_{G-1}$ for $\left[\check{U}_{G-1}, \check{U}_G\right]$, the maximum objective value of (17) in $\left[\check{U}_{G-1}, \check{U}_G\right]$, is larger than or equal to the one in $\left[\check{U}_G, \check{U}_{G+1}\right]$.

- **Property 6:**
  In Case 4, if $\check{U}_G \leq \frac{\sum_{i=G+1}^{d} \check{U}_i - \check{v}}{d+1-G} \leq \check{U}_{G+1}$ for $\left[\check{U}_G, \check{U}_{G+1}\right]$, then $\frac{\sum_{i=G}^{d} \check{U}_i - \check{v}}{d+1-(G-1)} \geq \check{U}_G$ for $\left[\check{U}_{G-1}, \check{U}_G\right]$ and $\frac{\sum_{i=G+2}^{d} \check{U}_i - \check{v}}{d+1-(G+1)}$ $\leq \check{U}_{G+1}$ for $\left[\check{U}_{G+1}, \check{U}_{G+2}\right]$, and the maximum value of (17) in the interval $\left[\check{U}_G, \check{U}_{G+1}\right]$ is larger than or equal to the ones in its neighbor intervals.

Properties 2-6 also apply for adjacent intervals $\left[-\check{v}, \check{U}_{L+1}\right]$ and $\left[\check{U}_{L+1}, \check{U}_{L+2}\right]$ in Case 3.

We omit the proof of Properties 1-6 since they are direct applications of 1-D quadratic optimization. Property 1 indicates that it is sufficient for the algorithm to start searching γ^* from Case 3. Properties 2 & 3 imply that, for some interval, if the axis of symmetry is on the right hand side of the interval, then one only needs to consider the intervals to the right. Similarly, Properties 4 & 5 indicate that, for some interval, if the axis of symmetry is on the left hand side of the interval, then one only needs to consider the intervals to the left. Property 6 combined with Properties 1-5 imply that, for certain interval, if it contains the axis of symmetry, then γ^* is the axis of symmetry point. Thus, we can draw the following conclusion: (1) if $\max\left(\check{U}_d, -v\right) < 0$, then

$$\gamma^* = 0;$$

(2) if $-\check{v} > \check{U}_d$, then

$$\gamma^* = \max(-\check{v}, 0);$$

(3) if $\check{U}_G \leq \frac{\sum_{i=G+1}^{d} \check{U}_i - \check{v}}{d+1-G} \leq \check{U}_{G+1}$ for a certain interval $\left[\check{U}_G, \check{U}_{G+1}\right]$, then

$$\gamma^* = \frac{\sum_{i=G+1}^{d} \check{U}_i - \check{v}}{d+1-G}.$$

At each move, the axis of symmetry $\frac{\sum_{i=G+1}^{d} \check{U}_i - \check{v}}{d+1-G}$ can be calculated by a constant operation based on the value from the last step, and the time complexity of searching γ^* reduces from quadratic to $\mathcal{O}(d\log(d))$ as the computation is dominated by the sorting operation. Once γ^* is determined, we can compute $\tilde{w}$ and $\widetilde{Q}$ by (18). Note that, the subproblem of the proximal operator associated with the convex relaxation in [2] is solved by searching for the dual variable in a different way with time complexity $\mathcal{O}(d^2)$.

In summary, we reformulate the proximal operator for the original weak hierarchical Lasso by factorizing the unknown coefficients. The reformulated proximal operator is shown to admit a closed form solution, which enables directly solving the weak hierarchical Lasso problem. Moreover, the subproblem of the proximal operator can be computed efficiently with a time complexity of $\mathcal{O}(d\log(d))$. The detailed algorithm for solving the proximal operator (12) is described in Algorithm 1. We give the details of eWHL algorithm in Algorithm 2. Following [15], we choose the step size $t^{(k)}$ by the Barzilai-Borwein (BB) Rule.

Algorithm 1 Computation of the Proximal Operator of Weak Hierarchical Lasso

```
Input: $v \in \mathbb{R}^{d\times 1}$, $U \in \mathbb{R}^{d\times d}$, $t \in \mathbb{R}_+$, $\lambda \in \mathbb{R}_+$
Output: $w \in \mathbb{R}^{d\times 1}$, $Q \in \mathbb{R}^{d\times d}$
$\check{v} = \text{sign}(v) \odot v - \frac{\lambda}{t}\mathbf{1}$;
$\check{U} = \text{sign}(U) \odot U - \frac{\lambda}{2t}\mathbf{1}\mathbf{1}^T$;
for $j = 1 : d$ do
    $c = -\check{v}_j$;
    Sort $\check{U}_{\cdot,j}$ to get a sequence $\mathcal{S}$ in ascending order where
    $\mathcal{S}_1 \leq \mathcal{S}_1 \leq \ldots \leq \mathcal{S}_d$;
    if $\mathcal{S}_d < 0$ and $c < 0$ then
        $\tilde{w}_j = \mathbf{0}$;
        $\check{Q}_{\cdot,j} = \mathbf{0}$;
    else
        if $\mathcal{S}_d < c$ then
            $\gamma = \max(c, 0)$;
        else
            $k = d$;
            while 1 do
                $c = c + \mathcal{S}_k$;
                $k = k - 1$;
                if $c/(d+1-k) \geq \mathcal{S}_k$ then
                    $\gamma = c$;
                    break;
                end if
            end while
        end if
        $\tilde{w}_j = \max(\check{v}_j + \gamma, \mathbf{0})$;
        $\widetilde{Q}_{\cdot,j} = \max\left(\check{U}_{\cdot,j} - \gamma, \mathbf{0}\right)$;
    end if
end for
$w = \text{sign}(v) \odot \tilde{w}$;
$Q = \text{sign}(U) \odot \widetilde{Q}$;
```

4. EXPERIMENTS

In this section, we evaluate the efficiency and effectiveness of the proposed algorithm on both synthetic and real data sets. In our first experiment, we compare the efficiency of our proposed algorithm and the convex relaxation of weak hierarchical Lasso [2] on synthetic data sets where the weak hierarchical structure holds between main effects and interaction effects. In our second experiment, we compare the classification performance of the weak hierarchical Lasso with other classifiers and sparse learning techniques on the data collected from Alzheimer's Disease Neuroimaging Initiative (ADNI)[1].

4.1 Efficiency and Effectiveness Comparison on Synthetic Data Sets

In this experiment, we compare the efficiency of the proposed eWHL algorithm with the convex relaxation on synthetic data sets. Our algorithm is built upon the GIST framework which is available online [16]. The source code of the convex relaxed weak hierarchical Lasso (cvxWHL) was available in the R package "*hierNet*" [3] where the optimiza-

[1] http://www.adni-info.org/

Algorithm 2 The Efficient Weak Hierarchical Lasso Algorithm (eWHL)

```
Input: $X \in \mathbb{R}^{n\times d}$, $Z \in \mathbb{R}^{n\times(d\cdot d)}$, $\lambda \in \mathbb{R}_{+}$, $\eta > 1$
Output: $w \in \mathbb{R}^{d\times 1}$, $Q \in \mathbb{R}^{d\times d}$
    Initialize $k \leftarrow 0$ and starting points $w^{(0)}$ and $Q^{(0)}$;
    repeat
        Choose the step size $t^{(k)}$ by the BB Rule
        repeat
            $v^{(k)} = w^{(k)} - \nabla_w \mathcal{L}\left(w^{(k)}, Q^{(k)}\right)/t^{(k)}$;
            $U^{(k)} = U^{(k)} - \nabla_Q \mathcal{L}\left(w^{(k)}, Q^{(k)}\right)/t^{(k)}$;
            Solve $\left(w^{(k+1)}, Q^{(k+1)}\right)$ by Algorithm 1 with input $\left(v^{(k)}, U^{(k)}, t^{(k)}, \lambda\right)$;
            $t^{(k)} \leftarrow \eta t^{(k)}$;
        until line search criterion is satisfied
        $k \leftarrow k+1$
    until stop criterion is satisfied
```

tion procedure was implemented by C. Since the proposed algorithm in this paper directly solves the non-convex weak hierarchical Lasso (5), and the eventual goal of the convex relaxed weak hierarchical Lasso is also to find a good "relaxed" solution to the original problem, we compare the two algorithms in terms of the objective function in (5). In the experiment, entries of $X \in \mathbb{R}^{n\times d}$ are i.i.d generated from the standard normal distribution, *i.e.,* $X_{i,j} \sim N(0,1)$. The matrix of interactions, Z, is then generated via the normalized X where $Z = \left[Z^{(1)}, Z^{(2)}, \ldots, Z^{(d)}\right]$, $Z^{(i)} \in \mathbb{R}^{n\times d}$, $Z^{(i)}_{\cdot,j} = X_{\cdot,i} \odot X_{\cdot,j}$. The ground truths $w \in \mathbb{R}^{d\times 1}$ and $Q \in \mathbb{R}^{d\times d}$ are generated based on the weak hierarchical structure $\|Q_{\cdot,j}\|_1 \leq |w_j|, j = 1, \ldots, d$. In addition, we vary the ratio of coefficient sparsity, *i.e.,* the portion of zero entries in w and Q, from 30% to 85%. Then, the outcome vector Y is constructed as $Y = Xw + \frac{1}{2}Z\cdot\mathrm{vec}(Q) + \epsilon$ where X and Z are normalized to zero mean and unit standard deviation and $\epsilon \sim N(\mathbf{0}, 0.01\cdot\mathbf{I})$. We use sample size $n = 100$ and 200 and we choose the number of main effects d from {100, 200, 300, 400, 500, 600}. The parameter of the l_1 penalty, λ, is chosen from {1, 3, 5, 10, 20}. All algorithms are executed on a 64-bit machine with Intel(R) Core(TM) quad-core processor (i7-3770 CPU @ 3.40 GHz) and 16.0 GB memory. We terminate the algorithm when the maximum relative difference of the coefficients between two consecutive iterations is less than $1e^{-5}$. We run 20 trials for each setting and report the average execution time. The detailed results are shown in Table 1.

From Table 1, we observe that eWHL is significantly faster than cvxWHL. Our algorithm is up to 25 times faster than the competing algorithm. As the dimension increases, the running time of cvxWHL increases much faster than our proposed algorithm. Specifically, when the number of individual features increases to 400 (corresponds to 80200 interactions), cvxWHL may take more than one thousand seconds, while the proposed eWHL is reasonably fast even when the number of total variables is around two hundred thousands.

To make further comparisons of efficiency, we randomly generate three synthetic data sets where the weak hierarchical structure between main effects and interactions holds. The three data sets are of the same sample size $n = 100$ and the number of individual features is $d = 300$. The ratios of zero entries in the ground truth are 85%, 60% and 30% respectively. The regularization parameters are chosen from {0.5, 1, 2, 4, 6, 8, 16, 32, 64}. On each data set, we first run cvxWHL, and then the objective value of (5) in the final step is recorded. Then, we run the proposed eWHL and terminate the algorithm when the objective value of (5) is less than the one obtained by cvxWHL. The running time and the number of iterations needed to achieve the same objective value of both algorithms are reported in Figure 1. We can observe from Figure 1 that the proposed eWHL is much faster than cvxWHL.

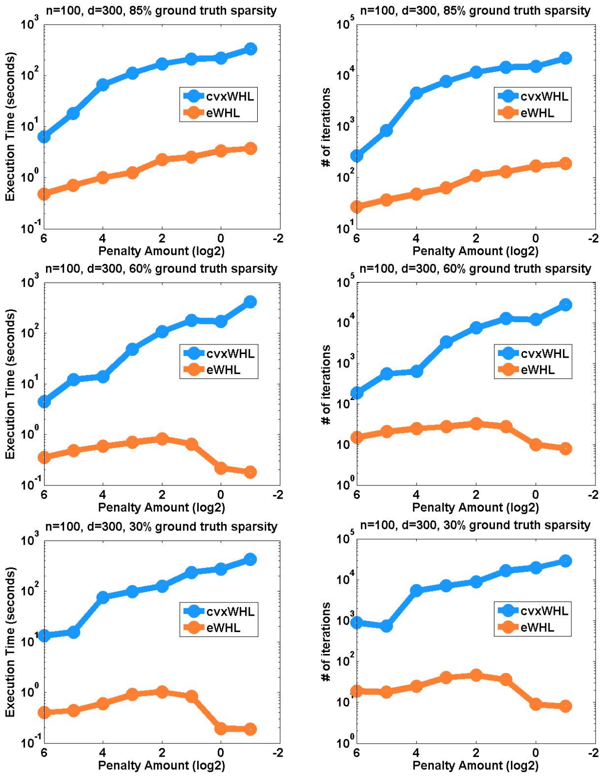

Figure 1: Comparison of the running time and the number of iterations by the two algorithms. Three synthetic data sets are generated where the portions of zeros in the ground truth are 85%, 60%, 30% respectively. The plots in the same row correspond to the same data set. The plots in the left column present the running time and those in the right column show the number of iterations.

Moreover, we also conduct an experiment to compare the recovery performance of eWHL and cvxWHL. We generate synthetic data sets with sample size $n = 100$ and the number of individual features is $d = 50$ (1225 cross interactions). The number of non-zero main effects varies from $\{3, 4, 5, 6, 7\}$ and the number of non-zero interaction effects is from $\{2, 4, 5, 8, 10\}$, respectively. For each setting, ten synthetic data sets are generated with noise $\epsilon \sim N(\mathbf{0}, 0.01\cdot\mathbf{I})$. We run both eWHL and cvxWHL with parameter selected via 5-fold cross-validation. Then we compute the sensitivity and specificity of recovery (where non-zero entries are positive and zero entries are negative). The means of sensitivity and specificity are plotted in Figure 2. We can observe that

Table 1: Comparison of execution time (second) of the proposed algorithm for the non-convex weak hierarchical Lasso (eWHL) and the one for the convex relaxed formulation (cvxWHL) on synthetic data. The penalty parameters used in the experiment are from $\{1, 3, 5, 10, 20\}$. The data is generated under the weak hierarchical constraints where the portion of sparse coefficients is controlled to 85%, 60% and 30%. Two sample sizes, $n = 100$ and $n = 200$, are used and we vary the number of individual features from $\{200, 300, 400, 500, 600\}$ corresponding to $\{20100, 45150, 80200, 125250, 180300\}$ interactions (including the self product terms).

		n = 100					n = 200				
85% Ground Truth Sparsity											
d	Methods	1	3	5	10	20	1	3	5	10	20
	cvxWHL	196.5536	54.8801	43.3018	27.2806	15.7909	116.8207	24.6601	17.8850	9.1765	4.7783
200	eWHL	15.9318	10.7613	7.2212	5.6287	2.6236	16.4134	9.5164	8.3827	5.4922	3.9255
	speedup	**12.3**	**5.1**	**6.0**	**4.8**	**6.0**	**7.1**	**2.6**	**2.1**	**1.7**	**1.2**
	cvxWHL	336.7086	213.7712	186.7997	109.7893	54.9521	319.6003	147.5044	112.5928	59.0820	36.2484
300	eWHL	35.6846	23.3044	17.9931	11.5569	10.8269	38.5045	20.0161	16.4566	10.3588	6.9153
	speedup	**9.4**	**9.2**	**10.4**	**9.5**	**5.1**	**8.3**	**7.4**	**6.8**	**5.7**	**5.2**
	cvxWHL	547.0450	280.6981	207.8486	170.4894	85.1425	921.7651	376.4949	256.8054	144.3066	81.4817
400	eWHL	52.8138	35.0482	29.5107	18.1944	13.8530	80.4882	54.1618	39.1673	26.6412	14.7667
	speedup	**10.4**	**8.0**	**7.0**	**9.4**	**6.1**	**11.5**	**7.0**	**6.6**	**5.4**	**5.5**
	cvxWHL	1018.9779	757.2096	524.9644	333.0070	204.2017	1405.9651	1142.2343	964.0598	286.2120	165.2558
500	eWHL	88.0526	66.0113	59.7805	42.2917	18.6453	127.0921	89.1293	70.0550	42.0014	29.0936
	speedup	**11.6**	**11.5**	**8.8**	**7.9**	**11.0**	**11.1**	**12.8**	**13.8**	**6.8**	**5.7**
	cvxWHL	2543.5021	1594.9729	1517.9605	887.8254	462.2604	2826.0083	1558.1431	1332.3515	873.6990	261.6806
600	eWHL	161.7944	100.3758	82.7961	71.1211	40.9529	197.5593	132.2163	107.5831	76.1834	45.0594
	speedup	**15.7**	**15.9**	**18.3**	**12.5**	**11.3**	**14.3**	**11.8**	**12.4**	**11.5**	**5.8**
60% Ground Truth Sparsity											
	cvxWHL	106.6262	40.3105	29.1357	20.8624	10.3064	113.3342	44.1169	27.2844	18.7616	11.9756
200	eWHL	15.1405	9.6837	6.9516	5.4949	3.3569	18.3514	10.5571	8.1777	5.1257	4.1127
	speedup	**7.0**	**4.2**	**4.2**	**3.8**	**3.1**	**6.2**	**4.2**	**3.3**	**3.7**	**2.9**
	cvxWHL	187.7983	131.7578	106.2882	61.3653	38.3189	290.0877	155.0435	131.7942	85.8886	44.4029
300	eWHL	33.3861	22.2763	16.3251	12.3395	9.2993	47.8122	26.1554	21.9835	13.7322	10.5702
	speedup	**5.6**	**5.9**	**6.5**	**5.0**	**4.1**	**6.1**	**5.9**	**6.0**	**6.3**	**4.2**
	cvxWHL	418.9647	276.2089	169.4631	131.9086	84.4169	686.8900	297.7161	226.6632	166.2235	85.4570
400	eWHL	66.8376	43.0676	35.7516	20.5413	11.9166	69.1413	41.3634	37.3495	25.9975	16.0270
	speedup	**6.3**	**6.4**	**4.7**	**6.4**	**7.1**	**9.9**	**7.2**	**6.1**	**6.4**	**5.3**
	cvxWHL	1501.3934	801.0146	548.8402	362.7110	206.0816	1333.8803	861.6311	729.1297	310.6121	202.0412
500	eWHL	112.5519	80.5276	60.2488	38.6862	25.5497	114.5243	73.6990	63.4899	47.9597	31.5058
	speedup	**13.3**	**9.9**	**9.1**	**9.4**	**8.1**	**11.6**	**11.7**	**11.5**	**6.5**	**6.4**
	cvxWHL	1976.0945	1733.8494	1814.1974	815.4298	323.4841	1622.9061	1205.8489	987.4595	1063.2823	333.8406
600	eWHL	159.8307	112.8232	80.3703	50.8628	34.4373	175.9730	140.7086	96.3447	73.7633	52.2213
	speedup	**12.4**	**15.4**	**22.6**	**16.0**	**9.4**	**9.2**	**8.6**	**10.2**	**14.4**	**6.4**
30% Ground Truth Sparsity											
	cvxWHL	139.4226	116.3866	85.1606	50.2425	23.8680	223.2468	165.9219	52.1613	40.8929	25.3084
200	eWHL	18.5023	13.6312	8.7261	7.1346	4.4546	20.8344	15.0214	9.7081	6.8616	4.0905
	speedup	**7.5**	**8.5**	**9.8**	**7.0**	**5.4**	**10.7**	**11.0**	**5.4**	**6.0**	**6.2**
	cvxWHL	275.4393	162.5627	139.4590	79.0609	41.2985	575.3758	223.1688	205.4368	142.9171	97.5106
300	eWHL	41.4815	27.6094	23.3748	14.2635	7.9047	52.3714	33.1059	25.5594	15.9962	10.6426
	speedup	**6.6**	**5.9**	**6.0**	**5.5**	**5.2**	**11.0**	**6.7**	**8.0**	**8.9**	**9.2**
	cvxWHL	916.5276	510.0533	342.0358	208.9260	104.5098	1688.7030	814.6646	560.1970	332.3118	204.4852
400	eWHL	75.6108	47.3789	38.8362	23.9130	17.6649	92.1627	60.4650	45.9561	34.6420	23.9751
	speedup	**12.1**	**10.8**	**8.8**	**8.7**	**5.9**	**18.3**	**13.5**	**12.2**	**9.6**	**8.5**
	cvxWHL	1460.8334	900.1424	767.7501	576.6498	242.9080	2003.7611	2040.7488	1632.3245	584.2366	313.8258
500	eWHL	114.9244	71.4278	58.4604	37.5124	25.9513	154.1934	102.7105	84.2729	63.9484	35.2531
	speedup	**12.7**	**12.6**	**13.1**	**15.4**	**9.4**	**13.0**	**19.9**	**19.4**	**9.1**	**8.9**
	cvxWHL	2799.5549	2842.9022	2076.6074	1148.0632	460.4660	4067.0519	2795.7589	2128.9981	1946.2750	1140.4244
600	eWHL	186.1483	119.8450	85.9816	65.2515	41.2607	165.9264	179.4403	146.7908	102.6978	71.8432
	speedup	**15.0**	**23.7**	**24.2**	**17.6**	**11.2**	**24.5**	**15.6**	**14.5**	**19.0**	**15.9**

both algorithms achieve high recovery rate while directly solving the original weak hierarchical Lasso leads to slightly better performance in recovering the non-zero effects.

4.2 Classification Comparison on ADNI Data

In this experiment, we compare the weak hierarchical Lasso with its convex relaxation as well as other classifiers on the Alzheimer's Disease Neuroimaging Initiative (ADNI) data set.

In Alzheimer's Disease (AD) research, Mild Cognitive Impairment (MCI) is an intermediate state between normal elderly people and AD patients [24]. The MCI patients are considered to be at high risk of progression to AD. Many recent work focus on how to accurately predict the MCI-AD conversion and identifying significant bio-markers for the prediction [8, 10, 12, 19, 21, 28, 31, 14].

In this experiment, we compare the classification performance of the proposed eWHL with the convex relaxation and other classifiers on the task of discriminating the MCI

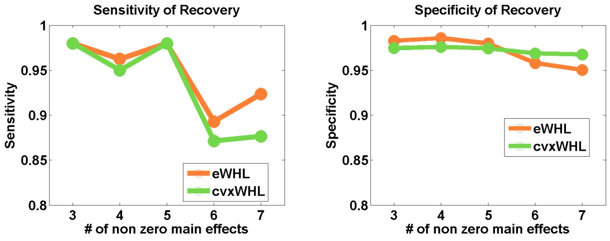

Figure 2: Comparison of eWHL and cvxWHL in terms of recovery on synthetic data sets.

Table 2: The performance of MCI converter vs. MCI non-converter classification achieved by random forest (RF), Support Vector Machine (SVM), Sparse Logistic Regression (spsLog), the convex relaxed weak hierarchical Lasso (cvxWHL) and the proposed algorithm (eWHL). Classifiers are performed on main effects only (top) and on both the main effects and interactions (bottom). The average and standard deviation of accuracy, sensitivity and specificity obtained from 10-fold cross-validation are reported.

Main Effects Only					
	RF	SVM	spsLog	cvxWHL	eWHL
Accuracy (%)	74.23 ± 8.67	75.22 ± 8.72	74.34 ± 9.56	NA	NA
Sensitivity (%)	78.75 ± 14.00	80.18 ± 13.89	80.18 ± 13.88	NA	NA
Specificity (%)	69.29 ± 11.63	69.76 ± 12.80	69.52 ± 13.74	NA	NA
Main Effects + Interactions					
	RF	SVM	spsLog	cvxWHL	eWHL
Accuracy (%)	71.26 ± 10.22	59.45 ± 14.43	73.57 ± 10.30	75.22 ± 11.02	77.42 ± 8.50
Sensitivity (%)	83.04 ± 13.18	59.29 ± 17.83	74.29 ± 16.22	75.71 ± 19.11	77.14 ± 12.05
Specificity (%)	58.10 ± 23.23	60.00 ± 15.42	72.86 ± 12.46	74.52 ± 16.84	77.62 ± 15.02

subjects who convert to dementia (*i.e.,* MCI converter) within a three-year period from the MCI subjects who remain at MCI (*i.e.,* MCI non-converter). The features used in the experiment (provided by our clinical collaborators) involve demographic information such as age, gender, years of education, clinical information such as scores of mini mental state examination (MMSE), Auditory Verbal Learning Test (A.V.L.T.), and the bio-markers including status of Apolipoprotein E, volume of hippocampus, thickness of Mid Temporal Gray Matter. There are 133 samples in total and the number of individual features is 36 (corresponds to 630 two way interactions). The interactions are generated by the normalized individual features and are normalized before entering the model. Since this is a classification task with binary labels, we replace the least square loss with logistic loss in the weak hierarchical Lasso. Besides the non-convex and convex weak hierarchical Lasso, we apply random forest (RF), Support Vector Machine (SVM) and sparse logistic regression on main effects, and on both main effects and interactions, respectively. We report the means and standard deviations of accuracy, sensitivity and specificity obtained from 10-fold cross-validation. The penalty parameters are tuned via 5-fold cross-validation in the training procedure. The sample statistics are shown in Table 3 and the classification performance is reported in Table 2.

Table 3: The statistics of the ADNI data set used in our experiment. The MCI converters (MCI-cvt) are characterized as positive samples and the MCI non-converters (MCI non-cvt) are used as negative samples.

	Total	(+) MCI-cvt	(-) MCI non-cvt
# of samples	133	71	62
# of main effects	36		
# of interactions	630		

From Table 2, we can observe that, if we only use individual features for classification, then all the classifiers are biased towards the positive class, *i.e.,* MCI converter. When interactions are included, we observe that the performances of random forest and SVM become worse. One possible reason is that the large number of variables brought by the interactions weakens their discriminative power. This is not the case for sparse logistic regression, which demonstrates the importance of feature selection. We can observe from the table that the convex relaxed weak hierarchical Lasso and the non-convex weak hierarchical Lasso achieve much better classification performance than the competitors. The improvement of the classification performance demonstrates the effectiveness of imposing hierarchical structures in interaction models. In addition, the superior classification performance (around 77% accuracy, sensitivity and specificity) of the proposed eWHL demonstrates that directly solving the non-convex weak hierarchical Lasso leads to solutions of higher quality than the convex relaxation.

5. CONCLUSIONS

In this paper, we propose an efficient algorithm, eWHL, to directly solve the non-convex weak hierarchical Lasso. One critical step in eWHL is to compute the proximal operator associated with the non-convex penalty functions. As one of our major contributions, we show that the proximal operator associated with the regularization function in weak hierarchical Lasso admits a closed form solution. Furthermore, we develop an efficient algorithm which computes each subproblem of the proximal operator with a time complexity of $\mathcal{O}(d \log d)$. Extensive experiments on both synthetic and real data sets demonstrate the superior performance of the proposed algorithm in terms of efficiency and accuracy.

In the future, we plan to apply the non-convex weak hierarchical Lasso to other important and challenging applications such as depression study [20]. In addition, we plan to extend the proposed techniques to solve the non-convex strong hierarchical Lasso formulation.

6. ACKNOWLEDGEMENT

This work was supported in part by NIH (R01 LM010730) and NSF (IIS-0953662, MCB-1026710, and CCF-1025177).

7. REFERENCES

[1] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[2] J. Bien, J. Taylor, and R. Tibshirani. A lasso for hierarchical interactions. *The Annals of Statistics*, 41(3):1111–1141, 2013.

[3] J. Bien and R. Tibshirani. *hierNet: A Lasso for Hierarchical Interactions*, 2013. R package version 1.5.

[4] R. J. Cadoret, W. R. Yates, G. Woodworth, M. A. Stewart, et al. Genetic-environmental interaction in the genesis of aggressivity and conduct disorders. *Archives of General Psychiatry*, 52(11):916, 1995.

[5] E. J. Candes and J. Romberg. Quantitative robust uncertainty principles and optimally sparse decompositions. *Foundations of Computational Mathematics*, 6(2):227–254, 2006.

[6] N. H. Choi, W. Li, and J. Zhu. Variable selection with the strong heredity constraint and its oracle property. *Journal of the American Statistical Association*, 105(489):354–364, 2010.

[7] A. d'Aspremont, L. El Ghaoui, M. I. Jordan, and G. R. Lanckriet. A direct formulation for sparse pca using semidefinite programming. In *NIPS*, volume 16, pages 41–48, 2004.

[8] C. Davatzikos, P. Bhatt, L. M. Shaw, K. N. Batmanghelich, and J. Q. Trojanowski. Prediction of mci to ad conversion, via mri, csf biomarkers, and pattern classification. *Neurobiology of aging*, 32(12):2322–e19, 2011.

[9] J. F. Dawson and A. W. Richter. Probing three-way interactions in moderated multiple regression: development and application of a slope difference test. *Journal of Applied Psychology*, 91(4):917, 2006.

[10] D. Devanand, G. Pradhaban, X. Liu, A. Khandji, S. De Santi, S. Segal, H. Rusinek, G. Pelton, L. Honig, R. Mayeux, et al. Hippocampal and entorhinal atrophy in mild cognitive impairment prediction of alzheimer disease. *Neurology*, 68(11):828–836, 2007.

[11] T. C. Eley, K. Sugden, A. Corsico, A. M. Gregory, P. Sham, P. McGuffin, R. Plomin, and I. W. Craig. Gene–environment interaction analysis of serotonin system markers with adolescent depression. *Molecular psychiatry*, 9(10):908–915, 2004.

[12] C. Fennema-Notestine, D. J. Hagler, L. K. McEvoy, A. S. Fleisher, E. H. Wu, D. S. Karow, and A. M. Dale. Structural mri biomarkers for preclinical and mild alzheimer's disease. *Human brain mapping*, 30(10):3238–3253, 2009.

[13] J. Gatt, C. Nemeroff, C. Dobson-Stone, R. Paul, R. Bryant, P. Schofield, E. Gordon, A. Kemp, and L. Williams. Interactions between bdnf val66met polymorphism and early life stress predict brain and arousal pathways to syndromal depression and anxiety. *Molecular psychiatry*, 14(7):681–695, 2009.

[14] P. Gong, J. Ye, and C. Zhang. Multi-stage multi-task feature learning. In *NIPS*, pages 1997–2005, 2012.

[15] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *ICML*, 2013.

[16] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye. *GIST: General Iterative Shrinkage and Thresholding for Non-convex Sparse Learning.* Tsinghua University, 2013.

[17] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani. *The elements of statistical learning*, volume 2. Springer, 2009.

[18] K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for large-scale 1-regularized logistic regression. *Journal of Machine learning research*, 8(7), 2007.

[19] H. Li, Y. Liu, P. Gong, C. Zhang, J. Ye, A. D. N. Initiative, et al. Hierarchical interactions model for predicting mild cognitive impairment (mci) to alzheimer's disease (ad) conversion. *PloS one*, 9(1):e82450, 2014.

[20] Y. Liu, Z. Nie, J. Zhou, M. Farnum, V. A. Narayan, G. Wittenberg, and J. Ye. Sparse generalized functional linear model for predicting remission status of depression patients. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, volume 19, pages 364–375. World Scientific, 2013.

[21] D. A. Llano, G. Laforet, and V. Devanarayan. Derivation of a new adas-cog composite using tree-based multivariate analysis: prediction of conversion from mild cognitive impairment to alzheimer disease. *Alzheimer Disease & Associated Disorders*, 25(1):73–84, 2011.

[22] D. C. Montgomery, E. A. Peck, and G. G. Vining. *Introduction to linear regression analysis*, volume 821. Wiley, 2012.

[23] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 1(3):123–231, 2013.

[24] R. C. Petersen. Mild cognitive impairment clinical trials. *Nature Reviews Drug Discovery*, 2(8):646–653, 2003.

[25] P. Radchenko and G. M. James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *Journal of the American Statistical Association*, 105(492):1541–1553, 2010.

[26] M. J. Somers. Organizational commitment, turnover and absenteeism: An examination of direct and interaction effects. *Journal of Organizational Behavior*, 16(1):49–58, 1995.

[27] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[28] J. Ye, M. Farnum, E. Yang, R. Verbeeck, V. Lobanov, N. Raghavan, G. Novak, A. DiBernardo, V. A. Narayan, et al. Sparse learning and stability selection for predicting mci to ad conversion using baseline adni data. *BMC neurology*, 12(1):46, 2012.

[29] M. Yuan, V. R. Joseph, and H. Zou. Structured variable selection and estimation. *The Annals of Applied Statistics*, pages 1738–1757, 2009.

[30] P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *The Annals of Statistics*, 37(6A):3468–3497, 12 2009.

[31] J. Zhou, J. Liu, V. A. Narayan, and J. Ye. Modeling disease progression via multi-task learning. *NeuroImage*, 78:233–248, 2013.

[32] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.