Feature Selection at the Discrete Limit

Miao Zhang[1], **Chris Ding**[1], **Ya Zhang**[2], **Feiping Nie**[1]
[1]University of Texas at Arlington, Texas, USA, 76019
[2]Shanghai Jiao Tong University, Shanghai, China, 200240
[1]miao.zhang@mavs.uta.edu, chqding@uta.edu, feipingnie@gmail.com
[2]ya_zhang@sjtu.edu.cn

Abstract

Feature selection plays an important role in many machine learning and data mining applications. In this paper, we propose to use $L_{2,p}$ norm for feature selection with emphasis on small p. As $p \to 0$, feature selection becomes discrete feature selection problem. We provide two algorithms, proximal gradient algorithm and rank-one update algorithm, which is more efficient at large regularization λ. We provide closed form solutions of the proximal operator at $p = 0, 1/2$. Experiments on real life datasets show that features selected at small p consistently outperform features selected at $p = 1$, the standard $L_{2,1}$ approach and other popular feature selection methods.

Introduction

Feature selection is important in a lot of machine learning tasks. It is aiming at selecting a subset of a large number of features, and is one of the most important techniques for dealing with high-dimensional data. A lot of machine learning models for classification, clustering, and other tasks such as those in bioinformatics need to deal with high dimensional data. Using high dimensional data in these applications directly will cause higher computational cost, and features those are irrelevant and redundant may also harm the performance of classification and clustering.

There are many research work on feature selection these years. Roughly, those feature selection methods can be categorized into 3 categories: wrapper methods (Kohavi and John 1997), filter methods (Langley 1994) and embedded methods. Previous research work on wrapper methods includes correlation-based feature selection (CFS) (Hall and Smith 1999) and support vector machine recursive feature elimination (SVM-RFE) (Guyon et al. 2002), etc. Research work on second category includes reliefF (Kira and Rendell 1992) (Kononenko 1994) (Kong et al. 2012), F-statistic (Ding and Peng 2003), mRMR (Peng, Long, and Ding 2005) and information gain (Raileanu and Stoffel 2000), etc. The goal of embedded methods is to maximize the margins between different classes, such as SVM-recursive feature elimination (Guyon et al. 2002), in which, features are removed iteratively based on some criteria. Embedded feature selection methods embed the selection process in the training process. Feature selection can be applied to both supervised and unsupervised learning, we focus here on the problem of supervised learning (classification). Methods mentioned above often have good performance, but also with high computational cost.

Sparsity regularization techniques recently have drawn a lot of attention in the studies of feature selection. For example, Bradley et al. (Bradley and Mangasarian 1998) and Fung et al. (Fung and Mangasarian 2000) proposed L_1-SVM method to do feature selection using L_1-norm regularization. Ng (Ng 2004) proposed logistic regression with L_1 norm regularization to do feature selection. Another proposed method uses both L_1-norm and L_2-norm to form a more structured regularization in (Wang, Zhu, and Zou 2007). Obozinsky et al.(Obozinski and Taskar 2006) and Argyriou et al.(Argyriou, Evgeniou, and Pontil 2007) developed a model with $L_{2,1}$-norm regularization to select features shared by multi tasks. Nie et al. (Nie et al. 2010) employed joint $L_{2,1}$-norm minimization on both loss function and regularization. There are also some other research work, such as (Kong and Ding 2013) (Chang et al. 2014).

In this paper, we propose to use $L_{2,p}$-norm regularization for feature selection with emphasis on small p. As $p \to 0$, feature selection becomes discrete feature selection problem. We provide two algorithms, proximal gradient algorithm and rank-one update algorithm to solve this discrete selection problem. The rank-one update algorithm is more efficient at large regularization λ. Experiments on real life datasets show that features selected at small p consistently outperform features selected by standard $L_{2,1}$ norm and other popular feature selection methods.

Feature Selection

In many applications, linear regression is used as the classification method. Using this method, feature selection can be done in many studies and experiments indicates that features selected using this approach work well using other classification methods (Obozinski and Taskar 2006) (Argyriou, Evgeniou, and Pontil 2007) (Nie et al. 2010) (Naseem, Togneri, and Bennamoun 2010).

Suppose $X = \{x_1, x_2, \cdots, x_n\}$ is the original dataset with d features for n data points, with associated class labels $Y =$

$\{y_1, y_2, \cdots, y_n\}$, and there are c classes. The feature selection problem is to select q features, i.e., q rows of X denoted as X_q, such that

$$J_0(X_q) = \min_W \|Y - W^T X_q\|_F^2 = \mathrm{Tr}\Big[Y^T Y - Y^T Y X_q^T (X_q X_q^T)^{-1} X_q\Big] \quad (1)$$

is minimized. We call $J_0(X_q)$ as *residual* of the selected features.

Clearly, this is a discrete optimization problem. Let $D = \{1, 2, \cdots, d\}$ denote the dimensions. we select a subset $q \subset D$ of discrete dimensions with the desired dimension such that the residual is minimized:

$$\min_{q \subset D} J_0(X_q). \quad (2)$$

This type of discrete optimization needs to search $C_d^q = d!/q!(d-q)!$ feature subsets, which is in general NP-hard problem. [Here to simplify the notation, q denotes either a *subset of dimensions* D, or the size of the selected subset.]

Instead of solving the discrete selection problem, we solve the following regularized problem,

$$\min_W \|Y - W^T X\|_F^2 + \lambda \|W\|_{2,p}^p \quad (3)$$

where $L_{2,p}$ norm on W is defined as

$$\|W\|_{2,p} = \Big(\sum_{i=1}^{d} \Big(\sum_{j=1}^{c} W_{ij}^2\Big)^{p/2}\Big)^{1/p} = \Big(\sum_{i=1}^{d} \|w^i\|^p\Big)^{1/p} \quad (4)$$

where w^i is the i-th row of W, d is the dimension of data and c is the number of classes in the problem. $L_{2,p}$ norm is the generalization of $L_{2,1}$-norm first introduced in (Ding et al. 2006).

Feature selection is achieved by increasing λ to certain value such that most of the rows of the optimal solution W^* become zeros. Clearly, if i-th row of W^* is zero, then the entire i-th row of X is not used in $\|Y - W^T X\|_F^2$. This means the i-th feature is eliminated. In other words, the nonzero rows of W indicate that these rows of X are selected as X_q. Thus the regularized formulation of Eq.(3) solves the discrete feature selection problem of Eq.(2).

In this paper, we investigate feature selection at different p values: $0 \le p \le 1$. The $p = 1$ case is the standard $L_{2,1}$ norm. As $p \to 0$, $\|W\|_{2,p}^p$ approaches the number of nonzero rows in W. Thus at small p, $\|W\|_{2,p}^p$ is more appropriate for feature selection.

One drawback of this approach is that $\|W\|_{2,p}^p (p < 1)$ is non-convex, so that we can not always find the exact global optimal solution. But in experiments, good local optimal solution can be computed and the selected features perform better than those selected at $p = 1$. This is a key finding of this paper (see detailed discussion in experiment section).

In the following of this paper, we present two computational algorithms - proximal gradient algorithm and rank-one update method to solve Eq.(3), and perform experiments on several real life datasets to demonstrate that $\|W\|_{2,p}^p$ at small p is a more effective feature selection method.

To validate the effectiveness of the selected q features, one measure is to utilize $J_0(X_q)$. If the selected q features are good, $J_0(X_q)$ should be small. Our experiments on several real life data indicates that features selected at p = 0.7, p = 0.5, p = 0.1 and p = 0, always have a smaller $J_0(X_q)$ as compared to features selected at $p = 1$. This indicates the validity of the proposed model of Eq.(3).

Proximal Gradient Algorithm

In this section, we apply the proximal gradient algorithm to solve our optimization problem in Eq.(3),

$$\min_W J(W) = f(W; X) + \lambda \|W\|_{2,p}^p$$

where X is the input data and f is supposed to be a convex function and its gradient is Lipschitz continuous. The gradient of f is Lipschitz continuous if $\|\nabla f(W_1) - \nabla f(W_2)\|_F \le \eta \|W_1 - W_2\|_F$, for any $W_1, W_2 \in \mathbb{R}^{d \times c}$, where η is a constant. More detailed introduction of proximal methods can be found in (Jenatton et al. 2010; Ji and Ye 2009; Nesterov 2007).

Starting with W_0, W is updated through iterations of $W_1, W_2, \cdots, W_t$. The important building block of proximal gradient algorithm is to solve the proximal operator equation

$$W_{t+1} = \arg\min_W \frac{1}{2\eta} \|W - A\|_F^2 + \lambda \|W\|_{2,p}^p \quad (5)$$

where $A = W_t - \eta \nabla f(W_t)$ represents the attempted update of W_t. Let w^i be the i-th row of W and Let a^i be the i-th row of A. It is easy to see the RHS of Eq.(5) can be written as

$$\sum_{i=1}^{d} \Big(\frac{1}{2}\|w^i - a^i\|^2 + \beta \|w^i\|^p\Big) \quad (6)$$

where $\beta = \eta\lambda$. Thus different rows of W can be independently computed. The optimization problem becomes solving the vector optimization problem

$$\min_w \frac{1}{2}\|w - a\|^2 + \beta \|w\|^p \quad (7)$$

where $w \in \mathbb{R}^d$, $a \in \mathbb{R}^d$.

One contribution of this paper is to study the optimal solution of this problem with $0 \le p \le 1$. (When $p > 1$ this problem is differentiable and easy to solve.)

(A) Since sparsity of the solution is a key issue, we derive the conditions for the optimal solution to be **completely sparse**, i.e., every components of the d-dimensional vector w is zero.

Theorem 1 *The optimal solution of the proximal operator optimization of Eq.(7) is completely sparse (i.e., $w = 0$) when*

$$\beta \ge \frac{1}{p} \frac{(1-p)^{(1-p)}}{(2-p)^{(2-p)}} \|a\|^{2-p}, \ 0 < p \le 1. \quad (8)$$

In particular, we have

$$p = 1 \quad : \quad \beta \ge \|a\| \quad (9)$$

$$p = \tfrac{1}{2} \quad : \quad \beta \ge \sqrt{16/27}\|a\|^{3/2} \quad (10)$$

$$p = 0 \quad : \quad \beta > (1/2)a_1^2 \quad (11)$$

where a_1 is the largest absolute value element of a.

The proof of Eq.(8) is provided in supplementary material (Zhang, Ding, and Zhang 2014) due to space limit. Proofs for the special cases Eqs.(9,10,11) are given in this paper.

(B) This optimization has closed form solutions at 3 special cases of Eqs.(9-11). The detailed analysis and computational algorithm of these closed form solutions are given in section 4. Eqs.(9-11) are proved there. One can easily verify that Eqs.(9,10) are the same as Eq.(8). We also present computational algorithm in section 4 when p is not one of these 3 cases.

The proximal gradient algorithm of this section is one approach to solve the feature selection problem Eq.(3). Here the proximal operator optimization of Eq.(5) (which reduces to Eq.(7)) is the critical step.

The rank-one update algorithm presented in next section is another approach to solve the feature selection problem Eq.(3). The proximal operator optimization of Eq.(7) discussed here is also the critical step in solving the rank-one update algorithm.

A Rank-one Update Algorithm

Another contribution of this paper is to introduce here a rank-one update algorithm to solve the feature selection problem of Eq.(3).

A problem with proximal gradient descent algorithm is that it converges slowly at large λ. This is an critical issue because in the feature selection problem, the **desired** optimal solution of W contains mostly zero rows because the selected features are typically much smaller than the number of original features. This occurs at large λ.

For example in the DNA and gene expression data, the number of features (genes) are typically 2000-4000, while we typically select 10 genes. This means that out of several thousands of rows of W, only 10 rows are nonzero. This happens at large λ values.

In this section, we propose a new algorithm - a rank-one update algorithm to solve the regularized formulation of Eq.(3). This algorithm is particularly efficient at large λ.

This can be explained fairly easily using the concrete example of multi-class feature selection.

$$\min_W \ \|Y - W^T X\|_F^2 + \lambda \|W\|_{2,p}^p \tag{12}$$

Let the column vector x^i contains the i-th row of X: $X^T = (x^1 \cdots x^d)$. We can decompose $W^T X = \sum_{i=1}^d w^i x^{iT}$ and $\|W\|_{2,p}^p = \sum_{i=1}^d \|w^i\|^p$. We can thus write the first term of Eq.(12) as

$$\|Y - W^T X\|_F^2 = \|(Y - \sum_{i \neq r} w^i x^{iT}) - w^r x^{rT}\|_F^2 = \|Y_r - w^r x^{rT}\|_F^2 \tag{13}$$

Thus the updating of r-th row of W can be written as

$$\min_{w^r} \ \|Y_r - w^r x^{rT}\|_F^2 + \lambda \|w^r\|^p \tag{14}$$

This optimization can be reduced to the proximal operator of Eq.(7). We have the following proposition

Proposition 2 *The optimization of Eq.(14) is identical with the optimization of Eq.(7) with the correspondence:*

$$w = w^r, \ b = Y_r^T x^r / \|x^r\|^2, \ \beta = \lambda / (2\|x^r\|^2). \tag{15}$$

Proof. We expand

$$\begin{aligned} \|Y_r - w^r x^{rT}\|_F^2 &= \mathrm{Tr}(x^r w^{rT} w^r x^{rT} - 2 Y_r^T w^r x^{rT} + Y_r^T Y_r) \\ &= \mathrm{Tr}(x^{rT} x^r w^{rT} w^r - 2 (Y_r x^r)^T w^r + Y_r^T Y_r) \\ &= \|x^r\|^2 \|w^r - b\|^2 - \|x^r\|^2 \|b\|^2 + \|Y_r\|^2 \end{aligned}$$

with b given in Eq.(15). Ignoring the last 2 terms which are independent of w^r, we have

$$\|Y_r - w^r x^{rT}\|_F^2 + \lambda \|w^r\|^p = 2\|x^r\|^2 \left(\frac{1}{2}\|w^r - b\|^2 + \beta \|w^r\|^p\right) \tag{16}$$

where with β given in Eq.(15). The positive constant $\|x^r\|^2$ drops out in the optimization. This completes the proof. □

The complete algorithm for rank-one update is presented in Algorithm 1, and updating a single row of W will be solved in section 4.1 - 4.3.

Algorithm 1 Rank-one Update Algorithm

Input: X, Y, W_0
parameters λ, p in $L_{2,p}$ norm
Output: W
Procedure:
$W = W_0$
while W not converged **do**
  for $r = 1$ to d **do**
    $Y_r = Y - \sum_{i \neq r} w^i x^{iT}$
    $b = Y_r^T x^r / \|x^r\|^2$
    $\beta = \lambda / (2\|x^r\|^2)$
    $w^r \longleftarrow \arg\min_{w^r} \frac{1}{2}\|w^r - b\|^2 + \beta \|w^r\|^p$
    switch p
      case 1: $p = 1$, standard $L_{2,1}$ norm
      case 2: $p = 0.5$, solve w^r using Eq.(23)
      case 3: $p = 0$, solve w^r using Eq.(20)
      case 4: $0 < p < 1, p \neq 0.5$, solve w^r using Eq.(25)
  end for
end while
Output W

This technique is called *rank-one update* technique, and popularly used in matrix computation.

Analysis of Proximal Operator

Solving the proximal operator equation Eq.(7) is the key step in both the proximal gradient algorithm (section 2) and the rank-one update algorithm (section 3). In this section, we present detailed analysis of the optimization problem of Eq.(7) for various cases of p.

First we transform the vector optimization problem of Eq.(7) into a scalar optimization problem. We have

Proposition 3 *The solution of Eq.(7) can be formulated as*

$$w = za, \ z \in \mathcal{R}, \ z \geq 0$$

and z is obtained by solving

$$\min_{z \geq 0} \ f(z) = \frac{1}{2}(z-1)^2 + \sigma z^p \tag{17}$$

where $\sigma = \beta \|a\|^{p-2} = \eta \lambda \|a\|^{p-2}$.

Proof. Let $w = (w_1 \cdots w_d)$ and $a = (a_1 \cdots a_d)$. From Eq.(7), it is clear that components of w must has the same sign as components of a. Second, if we express w as an unit direction $\hat{w}$ multiplying the magnitude $\|w\|$, i.e., $w = \hat{w}\|w\|$, the unit direction $\hat{w}$ must be in the same direction of a; this is because the penalty term $\|w\|^p$ is independent of the unit direction $\hat{w}$, and the first term $\|w - a\|^2$ is minimized when $\hat{w} = \hat{a} \equiv a/\|a\|$. This implies $w_i = za_i,\ i = 1 \cdots d$. Substituting $w = za$ into Eq.(7) leads to

$$\min_z \ \frac{1}{2}\|a\|^2(z-1)^2 + \eta\lambda\|a\|^p z^p \tag{18}$$

which is identical to Eq.(17). □

Therefore, to solve the proximal operator Eq.(7) is reduced to solve Eq.(17).

In the following, we solve the proximal operator in Eq.(7) or Eq.(17) when $0 \le p \le 1$, which is identical to solve Eq.(14) in rank-one update algorithm.

At $p = 1$, from Eq.(17), the KKT complementarity slackness condition $(\partial J/\partial z)z = 0$ gives $(z - 1 + \sigma)z = 0$. Thus the solution is $z = \max(1 - \sigma, 0)$. This also proves the sparse condition in Theorem 1. This result is known in earlier studies.

We present closed form solutions at $p = 0, p = 1/2$ below. To our knowledge, these results are not known previously.

Closed Form Solution at $p = 0$

When $p = 0$, we solve the optimization problem of Eq.(7), which is written as the following,

$$\min_w \ J(w) = \frac{1}{2}\|w - a\|^2 + \beta\|w\|^0 \tag{19}$$

where $\|w\|^0$ means the number of nonzero elements in vector w and $\beta = \eta\lambda$.

Theorem 4 *The solution of Eq.(19) is the following,*

1) First, sort the absolute value of a in descending order, such that $|a_1| > |a_2| > \cdots > |a_n|$, suppose there are n elements in a;

2) The optimal solution of w is given by w^*,

$$w^* = \begin{cases} a, & \text{if} \quad \beta < \frac{1}{2}a_n^2 \\ (a_1, \cdots, a_{k-1}, 0, \cdots, 0), & \text{if } \frac{1}{2}a_k^2 < \beta < \frac{1}{2}a_{k-1}^2 \\ (0, 0, \cdots, 0), & \text{if} \quad \beta > \frac{1}{2}a_1^2 \end{cases} \tag{20}$$

Proof. In the following proof, we use w_k to denote a solution where the number of nonzero elements in the vector w is k. This is because once we know the number of nonzero elements in w, Eq.(19) is trivial to solve.

When $\beta = 0$, obviously $w^* = w_n = a$, and $J(w_n) = 0 + \beta n$. If $\beta = \infty$, $w^* = w_0 = 0$. Thus we let β increases from 0, which causes the number of nonzero elements in solution w^* decrease; we find the corresponding condition on β to have the correct number of nonzero elements in the solution.

As β increases slightly from $\beta = 0$, the number of nonzero elements in w^* will drop to $n - 1$ from n, i.e., $w^* = w_{n-1} = (a_1, a_2, \cdots, a_{n-1}, 0)$ and and $J(w_{n-1}) = \frac{1}{2}a_n^2 + \beta(n-1)$. For w_{n-1} to be the optimal solution, we need $J(w_{n-1}) \le J(w_n)$. This condition leads to the first condition in Eq.(20).

As β keeps increasing, the number of nonzero elements in w^* keeps decreasing. Because that given the number of nonzero elements in w^*, we can find the optimal solution to Eq.(19), we have the following useful recursion relation

$$J(w_r) - J(w_{r-1}) = \frac{1}{2}a_r^2 - \beta.$$

So when $\frac{1}{2}a_k^2 < \beta < \frac{1}{2}a_{k-1}^2$, we assert that $w^* = w_{k-1}$. This is because this condition implies

$$\frac{1}{2}a_n^2 < \cdots < \frac{1}{2}a_{k+1}^2 < \frac{1}{2}a_k^2 < \beta < \frac{1}{2}a_{k-1}^2 < \cdots < \frac{1}{2}a_1^2.$$

From this, use the above recursion relation, we have

$$J(w_n) > \cdots > J(w_k) > J(w_{k-1}) < J(w_{k-2}) < \cdots < J(w_1).$$

Thus $J(w_{k-1})$ is the lowest value.

As we further increase β to the condition $\beta > \frac{1}{2}a_1^2$, we have $w^* = (0, 0, \cdots, 0)$, which proves the sparse condition in Theorem 1. This completes the proof of Theorem 4. □

Illustration of the proof. We now demonstrate the 3 cases of the above proof using one simple example. Let $a = (6, 5, 4, 3, 2, 1)$ in Eq.(19). We seek optimization at different β. For simplicity, a has been properly sorted. Note $a_1 = 6, a_2 = 5, \cdots, a_6 = 1$. The optimal solutions are:
(a) $\beta = 0.1$. In this case $\beta < \frac{1}{2}a_6^2 = 0.5$, $w^* = a = (6, 5, 4, 3, 2, 1)$;
(b) $\beta = 5$. In this case, $4.5 = \frac{1}{2}a_4^2 < \beta < \frac{1}{2}a_3^2 = 8$, $w^* = (6, 5, 4, 0, 0, 0)$;
(c) $\beta = 19$. In this case, $\beta > \frac{1}{2}a_1^2 = 18$, $w^* = (0, 0, 0, 0, 0, 0)$. All these cases can be easily verified by direct computation.

Closed Form Solution at $p = 1/2$

Here we give the closed form solutions of Eq.(17) and also prove that sparsity condition in Theorem 1 when $p = 1/2$. Setting derivative of Eq.(17) equal to zero, we obtain

$$z - 1 + \frac{1}{2}\sigma z^{-1/2} = 0 \tag{21}$$

We are looking for solutions to this equation with $z \ge 0$. Let $z^{1/2} = y$ and $\mu = \frac{1}{2}\sigma$, we need to solve

$$y^3 - y + \mu = 0, \quad s.t. \quad y \ge 0 \tag{22}$$

The analytic solution of this cubic equation can be written in closed form. What is interesting here is that although the final solution a real number, the arithmetic uses complex numbers in the intermediate steps. The nonzero imaginary parts cancel out exactly at the end.

The solution is obtained by setting $y = s - t$ and compute s, t from

$$s^3 = -\frac{\mu}{2} + \sqrt{\frac{\mu^2}{4} - \frac{1}{27}}, \quad t^3 = \quad \frac{\mu}{2} + \sqrt{\frac{\mu^2}{4} - \frac{1}{27}} \tag{23}$$

When $\mu^2/4 - 1/27 \ge 0$, s^3, t^3 are real, and their cubic roots are real Clearly the computed y is a real number. However, it is clear that $y = s - t < 0$, not in the desired range $y \ge 0$.

The other two roots are complex and do not fit. This gives the sparsity condition in Theorem 1.

When $\mu^2/4 - 1/27 < 0$, s^3, t^3 are complex. The cubic roots are readily computed by expressing Eq.(23) as

$$s^3 = \alpha_s \exp(i\theta_s), \quad t^3 = \alpha_t \exp(i\theta_t). \tag{24}$$

where $\alpha_s, \alpha_t, \theta_s, \theta_t$ are all real. We need only 2 roots of each:

$$s_1 = \alpha_s^{1/3} \exp(i\theta_s/3),\ s_3 = \alpha_s^{1/3} \exp(i(\theta_s/3 - 2\pi/3)),$$

$$t_2 = \alpha_t^{1/3} \exp(i(\theta_t/3 + 2\pi/3)),\ t_3 = \alpha_t^{1/3} \exp(i(\theta_t/3 - 2\pi/3))$$

It can be proved that one root of Eq.(22) is always negative, thus not feasible. The other two roots are $y_0 = s_3 - t_3$, $y_1 = s_1 - t_2$. Furthermore, $0 \le y_0 \le y_1 \le 1$. Also, $f(z)$ of Eq.(17) reaches minimum at $z_1 = y_1^2$. Thus y_1 is the desired root.

Note that although s_1, s_3, t_2, t_3 are complex with nonzero imaginary parts, the relevant imaginary parts always cancel exactly; thus y_0 and y_1 have zero imaginary parts.

We illustrate this method using one example for $\mu = 0.2$ in Eq.(22). We calculate

$$s_1 = 0.4394 + 0.3745i,\ s_3 = 0.1046 - 0.5678i,$$

$$t_2 = -0.4394 + 0.3745i,\ t_3 = -0.1046 - 0.5678i,$$

The roots are $y_0 = 0.2092$, $y_1 = 0.8788$. The imaginary parts cancel exactly. $f(z)$ reaches minimum at $z_1 = y_1^2$.

Numerical Solution at $0 < p < 1$

We use Newton's method to solve Eq.(17). Starting at $z_0 = 1$, we iterate

$$z_{t+1} = z_t - g'(z)/g''(z), \tag{25}$$

where $g'(z) = z - 1 + \sigma p z^{p-1}$, and $g''(z) = 1 - \sigma p(1-p)z^{p-2}$. This algorithm converges to the local minimum $z_1 > 0$ if Eq.(8) does not hold. With 20 iterations, the solution is accurate with an error less than 10^{-14}, close to the machine precision. To find the global solution, we need to compare this local minima with another possible local minima $z_2 = 0$, see the figure in supplementary material (Zhang, Ding, and Zhang 2014). We compute $f(z_1), f(z_2)$, and then pick the one with smaller $f(\cdot)$ value.

Experiments

To validate the performance of our $L_{2,p}$ feature selection method, we apply it on three data sets: DNA dataset, which belongs to the Statlog collection and used in (Hsu and Lin 2002), and two publicly available microarray datasets: the small round blue cell tumors (SRBCT) dataset (Khan et al. 2001) and the malignant glioma (GLIOMA) dataset (Nutt et al. 2003). The DNA dataset contains total 2000 samples in three classes, each sample has 180 features. The SRBCT dataset contains total 83 samples in four classes. Every sample in this dataset contains 2,308 gene expression values. The GLIOMA dataset (Nutt et al. 2003) contains total 50 samples in four classes. Every sample in this dataset contains 2,308 gene expression values. We compare the performance of our method $0 \le p < 1$ with $p = 1$ and five other feature selection methods - SVM-RFE (Guyon et al. 2002), ReliefF (Kira and Rendell 1992) (Kononenko 1994), mRMR (Peng, Long, and Ding 2005), F-Statistic method (Ding 2002) and Mutual information (Battiti 1994).

Table 1: Residual error of selected features/dimensions, $J_0(X_q)$ in Eq.(1), at different p values on DNA data.

different p values	$J_0(X_q)$ with different q				
	q = 10	q = 20	q = 30	q = 40	q = 50
p = 1	636.698	510.696	461.988	431.647	404.883
p = 0.7	**625.042**	506.120	**444.398**	**417.162**	399.332
p = 0.5	**621.652**	**487.288**	**443.824**	417.898	**396.654**
p = 0.1	**625.042**	496.834	449.808	417.351	**397.808**
p = 0	**621.652**	**492.564**	446.046	**416.487**	399.536
SVM-RFE	1277.375	913.397	825.018	728.313	673.689
ReliefF	761.147	603.404	578.482	561.334	542.042
mRMR	778.504	521.113	456.343	434.658	410.838
F-Statistic	778.504	516.454	457.828	433.836	412.241
Mutual Info	778.504	516.454	456.343	434.658	410.838

Table 2: Residual error of selected features on SRBCT data

different p values	$J_0(X_q)$ with different q				
	q = 10	q = 20	q = 30	q = 40	q = 50
p = 1	12.754	8.186	2.857	1.886	1.309
p = 0.7	9.205	4.033	2.392	**1.064**	0.634
p = 0.5	**8.314**	**3.244**	**1.839**	**0.976**	**0.375**
p = 0.1	10.848	**3.681**	**2.276**	1.833	**0.398**
p = 0	**8.173**	4.308	2.394	1.553	0.883
SVM-RFE	20.922	10.068	6.170	4.871	2.117
ReliefF	18.003	8.241	4.992	2.927	1.976
mRMR	14.831	5.976	4.208	2.637	1.942
F-Statistic	13.208	6.181	3.282	2.602	1.889
Mutual Info	19.044	12.915	4.974	2.905	1.884

Effectiveness of the Features Selected at Small p

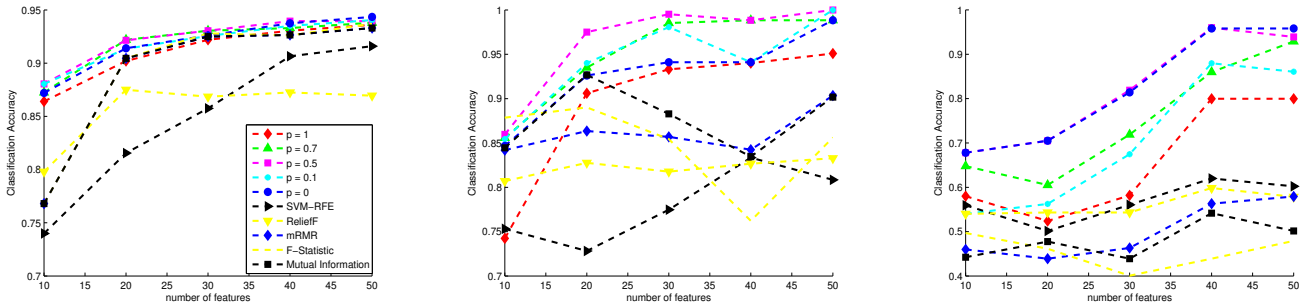
We compare our proposed feature selection methods in Eq.(3) at $p = 0, 0.1, 0.5, 0.7$, with the six methods mentioned above. SVM-RFE is original for 2-class. Here we implemented the multi-class extension (Zhou and Tuck 2007).

For each p value, we adjust λ such that the number of nonzero rows in W^* (optimal solution) is q. After we get the q features, we use the value of $J_0(X_q)$ in Eq.(1) to validate the effectiveness of the selected q features. if the q features are effective to represent the original X, the value of $J_0(X_q)$ should be small.

Table 1, 2, 3 give the value of $J_0(X_q)$ with different q features selected by different p value on dataset DNA, SRBCT and GLIOMA, respectively. As we can see, when $0 \le p < 1$, $J_0(X_q)$ is always smaller than that of $p = 1$ when the number of selected features q is fixed. Notice, on GLIOMA dataset, when $q = 50$, $J_0(X_q) = 0$, that is because the number of data points $n = 50$.

Classification Accuracy Comparison

For each dataset, we randomly split the dataset X into training set and testing set equally, ie, 50% of the data is training and 50% is testing. To get good statistics, we rerun these split process 20 times so splits are different from each run. And for each run, we adopt cross validation to ensure the fairness of the evaluation. Final results are the averages over these 20 runs. We use linear regression as our classifier. Fig-



(a) on DNA dataset (b) on SRBCT dataset (c) on GLIOMA dataset

Figure 1: Classification Accuracy by different feature selection methods on three dataset

Table 3: Residual error of selected features on GLIOMA

different p values	$J_0(X_q)$ with different q				
	q = 10	q = 20	q = 30	q = 40	q = 50
p = 1	14.520	8.822	4.045	1.511	0
p = 0.7	**14.442**	**4.853**	2.295	0.364	0
p = 0.5	**14.351**	**5.088**	1.670	**0.273**	0
p = 0.1	14.450	7.761	**1.662**	0.341	0
p = 0	**14.351**	**5.088**	**1.052**	**0.261**	0
SVM-RFE	23.888	19.914	11.856	5.515	0
ReliefF	22.093	15.151	8.069	3.812	0
mRMR	22.473	16.678	8.640	4.617	0
F-Statistic	24.1803	13.782	8.396	4.725	0
Mutual Info	25.3655	15.066	7.652	2.534	0

ure 1a 1b 1c illustrate the classification accuracies using different p values and on the 3 datasets mentioned above. As we can see, our proposed methods ($0 \leq p < 1$) outperform the previous method ($p = 1$) and other popular feature selection methods in most cases.

Running Time Comparison

Our optimization strategy is efficient, W converges quickly especially when λ is big (the number of selected features are small). Optimizing for $p < 1$ is just a little bit slower than $p = 1$, but can gain better results. We compare the running time of our rank-one update(RK1U) methods with the standard method on $L_{2,1}$ norm (Argyriou, Evgeniou, and Pontil 2007) - multi task learning algorithm (MTLA). Running time taken by different methods are listed in Table 4 (all methods converged to the same criteria: objective of Eq.(3) changes less than 10^{-6} between successive iterations), which shows that our methods are generally faster than the multi task learning method.

Discussion

As mentioned before, when $p < 1$, the optimization problem of Eq.(3) is non-convex, we cannot guarantee to find the global minima. But a good local minima can be found using reasonable initialization strategy. We initialize W in Eq.(7) using two methods: (1) ridge regression, i.e., replace the $L_{2,p}$ norm in Eq.(3) with Frobenius norm. This gives closed form solution for W. (2) global solution of W at $p = 1$. Our experiment results show that solutions of $p < 1$ are the best using these two initializations judged in terms of the objective of Eq.(1). For this reason, the residual (for fixed q) in Tables 1,2,3 at larger p values could be smaller than those at smaller p values. This is not inconsistent. This indicates, we believe, that the solution at these $p < 1$ is not necessarily the true global solution. This also reveals the *difficulty*, i.e., NP-hardness of this discrete optimization problem. Experiment results and theoretical arguments indicate that as p approaches 0, feature subset selected from Eq.(3) are better (smaller residual). This trend is clear but not strict in Tables 1,2,3. The key points in Table 1,2,3 are that $p < 1$ solutions (selected feature subsets) are better than $p = 1$ case and other popular feature selection methods.

Table 4: Running time (sec) on SRBCT and GLIOMA data sets. RK1U: our rank-one update algorithm.

Algorithm	SRBCT		GLIOMA	
	q = 20	q = 50	q = 20	q = 50
MTLA	473.7	1211.3	6332.0	10456.9
RK1U p = 1	42.7	88.9	1534.6	1762.0
RK1U p = 0.5	205.6	607.3	3561.9	3409.8
RK1U p = 0.1	219.2	679.8	3413.0	3350.9
RK1U p = 0	150.1	130.8	2005.6	1995.5

Summary

First, we propose to use $L_{2,p}$-norm regularization for feature selection with emphasis on small p. As $p \to 0$, feature selection becomes discrete feature selection problem. Second, we propose two efficient algorithms, proximal gradient algorithm and rank-one-update algorithm to solve this discrete feature selection problem. We provide substantial theoretical analysis and closed form solutions to the critical algorithmic part, the proximal operator. Extensive experiments on real life datasets show that features selected at small p consistently outperform other methods.

References

Argyriou, A.; Evgeniou, T.; and Pontil, M. 2007. Multi-task feature learning. In *NIPS*, 41–48.

Battiti, R. 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks* 5:537–550.

Bradley, P., and Mangasarian, O. L. 1998. Feature selection via concave minimization and support vector machines. In *Proc. International Conference of Machine Learning (ICML)*, 82–90.

Chang, X.; Shen, H.; Wang, S.; Liu, J.; and Li, X. 2014. Semi-supervised feature analysis for multimedia annotation by mining label correlation. In *Advances in Knowledge Discovery and Data Mining*, 74–85.

Ding, C., and Peng, H. 2003. Minimum redundancy feature selection from microarray gene expression data. In *J Bioinform Comput Biol*, 523–529.

Ding, C.; Zhou, D.; He, X.; and Zha, H. 2006. R1-PCA: Rotational invariant L1-norm principal component analysis for robust subspace factorization. *Proc. Int'l Conf. Machine Learning (ICML)*.

Ding, C. 2002. Analysis of gene expression profiles: Class discovery and leaf ordering. In *In Proc. 6th Int'l Conf. Research in Comp. Mol. Bio.(RECOMB 2002)*, 127–136. ACM Press.

Fung, G., and Mangasarian, O. L. 2000. Data selection for support vector machine classifiers. In *KDD*, 64–70.

Guyon, I.; Weston, J.; Barnhill, S.; and Vapnik, V. 2002. Gene selection for cancer classification using support vector machines. *Mach. Learn.* 46(1-3):389–422.

Hall, M. A., and Smith, L. A. 1999. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper.

Hsu, C.-W., and Lin, C.-J. 2002. A comparison of methods for multiclass support vector machines.

Jenatton, R.; Mairal, J.; Obozinski, G.; and Bach, F. 2010. Proximal methods for sparse hierarchical dictionary learning. In *ICML*, 487–494.

Ji, S., and Ye, J. 2009. An accelerated gradient method for trace norm minimization. In *ICML*.

Khan, J.; Wei, J. S.; Ringner, M.; Saal, L. H.; Ladanyi, M.; Westermann, F.; Berthold, F.; Schwab, M.; Antonescu, C. R.; Peterson, C.; and Meltzer, P. S. 2001. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat Med* 7:673–679.

Kira, K., and Rendell, L. A. 1992. A practical approach to feature selection. In *ML*, 249–256.

Kohavi, R., and John, G. H. 1997. Wrappers for feature subset selection. *ARTIFICIAL INTELLIGENCE* 97(1):273–324.

Kong, D., and Ding, C. H. Q. 2013. Efficient algorithms for selecting features with arbitrary group constraints via group lasso. In *ICDM*, 379–388.

Kong, D.; Ding, C. H. Q.; Huang, H.; and Zhao, H. 2012. Multi-label relieff and f-statistic feature selections for image annotation. In *CVPR*, 2352–2359.

Kononenko, I. 1994. Estimating attributes: Analysis and extensions of relief. 171–182.

Langley, P. 1994. Selection of relevant features in machine learning. In *In Proceedings of the AAAI Fall symposium on relevance*, 140–144.

Naseem, I.; Togneri, R.; and Bennamoun, M. 2010. Linear regression for face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 32(11):2106–2112.

Nesterov, Y. 2007. gradient methods for minimizing composite objective function. *CORE technical report*.

Ng, A. Y. 2004. Feature selection, l1 vs l2 regularization, and rotational invariance. In *ICML*.

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. Q. 2010. Efficient and robust feature selection via joint l2,1-norms minimization. In *NIPS*, 1813–1821.

Nutt, C. L.; Mani, D. R.; Betensky, R. A.; Tamayo, P.; Cairncross, J. G.; Ladd, C.; Pohl, U.; Hartmann, C.; McLaughlin, M. E.; Batchelor, T. T.; Black, P. M.; von Deimling, A.; Pomeroy, S. L.; Golub, T. R.; and Louis, D. N. 2003. Gene expression-based classification of malignant gliomas correlates better with survival than histological classification. *Cancer Research* 63:1602–1607.

Obozinski, G., and Taskar, B. 2006. Multi-task feature selection. Technical report, In the workshop of structural Knowledge Transfer for Machine Learning in the 23rd International Conference on Machine Learning (ICML).

Peng, H.; Long, F.; and Ding, C. 2005. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27:1226–1238.

Raileanu, L. E., and Stoffel, K. 2000. Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence* 41:77–93.

Wang, L.; Zhu, J.; and Zou, H. 2007. Hybrid huberized support vector machines for microarray classification. In *In ICML '07: Proceedings of the 24th International Conference on Machine Learning*, 983–990.

Zhang, M.; Ding, C.; and Zhang, Y. 2014. Supplementary material to 'Feature selection at the discrete limit (AAAI2014)'.

Zhou, X., and Tuck, D. P. 2007. Msvm-rfe: extensions of svm-rfe for multiclass gene selection on dna microarray data. *Bioinformatics* 23(9):1106–1114.