Direct Sparsity Optimization Based Feature Selection for Multi-Class Classification

Hanyang Peng[1], **Yong Fan**[2]
[1]National Laboratory of Pattern Recognition, Institute of Automaiton,
Chinese Academy of Sciences, 100190, Beijing, P.R. China
[2]Department of Radiology, Perelman School of Medicine, University of Pennsylvania,
Philadelphia, PA, 19104, USA
hanyang.peng@nlpr.ia.ac.cn, yong.fan@ieee.org

Abstract

A novel sparsity optimization method is proposed to select features for multi-class classification problems by directly optimizing a $\ell_{2,p}$-norm ($0 < p \leq 1$) based sparsity function subject to data-fitting inequality constraints to obtain large between-class margins. The direct sparse optimization method circumvents the empirical tuning of regularization parameters in existing feature selection methods that adopt the sparsity model as a regularization term. To solve the direct sparsity optimization problem that is non-smooth and non-convex when $0 < p < 1$, we propose an efficient iterative algorithm with proved convergence by converting it to a convex and smooth optimization problem at every iteration step. The proposed algorithm has been evaluated based on publicly available datasets. The experiments have demonstrated that our algorithm could achieve feature selection performance competitive to state-of-the-art algorithms.

1 Introduction

Feature selection has been an important component in machine learning models (Guyon & Elisseeff, 2003). Feature selection approaches in general can be divided into three groups: filter methods (Kira & Rendell, 1992; Lewis, 1992; Peng et al., 2005), wrapper methods (Guyon et al., 2002), and embedded methods (Cawley et al., 2006; Wang et al., 2008; Xiang et al., 2012). The filter methods use proxy measures that are independent on the machine learning models to rank features according to their relevance to the learning problem. The wrapper methods search subsets of features to optimize a given learning model's performance, and typically have higher computational cost than the filter methods. The embedded methods integrate the feature selection task into the model learning, and are able to achieve good performance with a moderate computational cost. Particularly, sparse linear model based feature selection methods have achieved promising performance. The sparse linear model based methods typically adopt ℓ_1-norm regularization, and many variants have been proposed with different sparsity regularization terms, such as Lasso (Tibshirani, 1996) and sparse support vector machine (SVM) (Bradley & Mangasarian, 1998; Mangasarian, 2006).

In multi-task learning, various $\ell_{2,1}$-norm (Liu et al., 2009; Nie et al., 2010; Obozinski et al., 2006) and $\ell_{\infty,1}$-norm (Liu et al., 2009) based regularization models have been investigated for selecting features with joint sparsity across different tasks. Moreover, group Lasso based methods (Kong & Ding, 2013; Kong et al., 2014) have also been proposed in recent years. In fact, the multi-task feature selection algorithms share similarities with group lasso based methods. Since non-convex ℓ_p-norm or $\ell_{r,p}$-norm ($0 < p < 1$) based regularization models can yield sparser solutions than ℓ_1-norm or $\ell_{r,1}$-norm based models, they have gained increasing attention in recent studies (Chartrand & Staneva, 2008; Liu et al., 2007; Zhang et al., 2014).

The sparse linear model based feature selection algorithms typically take a trade-off between a data-fitting loss function term and a sparsity term, therefore there inevitably exists residual in the loss function. However, little is known about such a residual's impact on the feature selection. Most of these algorithms are developed for classification problems. However, their data-fitting loss functions adopted are typically based on the regression of class labels, rather than surrogates of 0-1 loss that are more appropriate for classification problems (Bartlett et al., 2004). To overcome the limitations of existing sparse model based feature selection methods, we present a novel feature selection method via directly optimizing a linear model's sparsity with $\ell_{2,p}$-norm ($0 < p \leq 1$), subject to data-fitting inequality constraints, instead of adopting the sparsity as a regularization term. Our constrained optimization formulation circumvents the difficulty of regularization parameter setting, and the inequality constraint based data-fitting loss function enables large between-class margins, similar to SVMs, but capable of handling multi-class problems. We propose an efficient algorithm to solve the optimization problem associated with the direct sparsity optimization, which is non-convex and non-smooth

when $0 < p < 1$, by transforming it to a Frobenius-norm induced problem at each iteration step, which has been proved to converge to Karush–Kuhn–Tucker (KKT) points.

The proposed algorithm has been evaluated based on 6 publicly available datasets, and extensive comparison experiments have demonstrated that our algorithm could achieve feature selection performance competitive to state-of-the-art algorithms, including ReliefF (Kira & Rendell, 1992), Min-Redundancy Max-Relevance (mRMR) (Peng, et al., 2005), ℓ_1-norm based ℓ_1-SVM (Bradley & Mangasarian, 1998; Mangasarian, 2006), and $\ell_{2,1}$-norm based Robust Feature Selection (RFS) (Nie, et al., 2010; Xiang, et al., 2012).

2 Direct Sparsity Optimization Based Feature Selection (DSO-FS)

Throughout this paper, matrices are written in bold uppercase, vectors are written in bold lowercase, and all the scalars are denoted by normal letters. $\boldsymbol{I}$ denotes an identity matrix and $\mathbf{1}$ denotes a vector or matrix with all the elements equal to 1. Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, the $\ell_{r,p}$-norm$(r > 0, p > 0)$[1] of $\boldsymbol{A}$ is defined as:

$$\|\boldsymbol{A}\|_{r,p} = \left(\sum_{i=1}^{m}\left(\sum_{j=1}^{n}\left|\boldsymbol{A}_{i,j}\right|^{r}\right)^{\frac{p}{r}}\right)^{\frac{1}{p}} = \left(\sum_{i=1}^{m}(\|\boldsymbol{a}_i\|_r)^p\right)^{\frac{1}{p}}, \quad (1)$$

where $\|\boldsymbol{a}_i\|_r$ denotes ℓ_r-norm of the i-th row vector of $\boldsymbol{A}$.

Given m training samples $\{\boldsymbol{x}^i, y^i\}_{i=1}^m$ where $\boldsymbol{x}^i \in \mathbb{R}^n$ is a data point and y^i is its associated class label in $c(c \geq 2)$ classes, the multiclass classification problem can be modeled as a linear learning problem. For simplicity, the bias of the standard linear regression is absorbed into $\boldsymbol{W}$ as an additional dimension with all elements equal to 1.

$$\boldsymbol{XW} = \boldsymbol{Y}, \quad (2)$$

where $\boldsymbol{X} = [\boldsymbol{x}^1; \boldsymbol{x}^2; \dots; \boldsymbol{x}^m; 1]$, $\boldsymbol{W} \in \mathbb{R}^{(n+1)\times c}$ is the weight matrix to be learned, and $\boldsymbol{Y} = [\boldsymbol{f}^1; \boldsymbol{f}^2; \dots, \boldsymbol{f}^i; \dots; \boldsymbol{f}^m] \in \mathbb{R}^{m\times c}$ is a class label matrix with labels rearranged using a one-versus-rest model, i.e., $\boldsymbol{f}^i = [-1, \dots, 1, \dots, -1] \in \mathbb{R}^c$ (the j-th element is 1 and others

For selecting a subset of features, $\boldsymbol{W}$ should have sparse columns and share joint sparsity along its row direction since each row of $\boldsymbol{W}$ corresponds to the same feature. Therefore, we model the feature selection problem as

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_{2,0}, \ \ s.t., \boldsymbol{XW} = \boldsymbol{Y}. \quad (3)$$

where $\|\mathbf{W}\|_{2,0}$ is the number of non-zero rows in $\boldsymbol{W}$, of which not all the elements are zero.

It is NP-hard to solve the optimization problem of Eqn. (3). Therefore $\ell_{2,p}$-norm $(0 < p \leq 1)$ can be adopted instead, resulting in a relaxed sparsity optimization problem:

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_{2,p}, \ \ s.t., \boldsymbol{XW} = \boldsymbol{Y}, \quad (4)$$

where $0 < p \leq 1$.

Many studies (Liu, et al., 2009; Nie, et al., 2010; Obozinski, et al., 2006) assumed that the $\ell_{2,1}$-norm based problems is equivalent to $\ell_{2,0}$-norm based problems under certain conditions (Candes & Tao, 2005). In practice, $\ell_{2,p}$-norm $(0 < p < 1)$ can lead to sparser solutions in most cases although it is non-convex (Chartrand & Staneva, 2008; Liu, et al., 2007; Zhang, et al., 2014).

It is desired that the classification model's margins between classes are as large as possible for obtaining improved generalization performance. Accordingly, the equality constraint in Eqn. (4) is relaxed to be inequality constraints, *i.e.*,

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_{2,p}, \ \ s.t., \boldsymbol{Y} \odot \boldsymbol{XW} \geqslant \mathbf{1}, \quad (5)$$

where $\mathbf{1} \in \mathbb{R}^{m\times c}$, $\odot$ is a Hadamard product operator for element-wise multiplication between two matrices of the same dimensions, and $\geqslant$ denotes that the elements of the left matrix are greater than or equal to their corresponding ones of the right matrix.

The optimization problem of Eqn. (5) can be reformulated by introducing a slack variable $\boldsymbol{E} \in \mathbb{R}^{m\times c}$ whose elements have the same positive or negative sign as the corresponding elements of $\boldsymbol{Y}$, as

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{E}} \ & \|\boldsymbol{W}\|_{2,p} \\ s.t. \ & \boldsymbol{XW} = \boldsymbol{Y} + \boldsymbol{E} \\ & \boldsymbol{Y} \odot \boldsymbol{E} \geqslant \mathbf{0}, \end{aligned} \quad (6)$$

where $\boldsymbol{E} \in \mathbb{R}^{m\times c}$.

To solve the optimization problem of Eqn. (6), we first solve the linear equation $\boldsymbol{XW} = \boldsymbol{Y} + \boldsymbol{E}$ to obtain the solution space of $\boldsymbol{W}$, and then directly search the solution space to find a solution with the minimum of $\|\boldsymbol{W}\|_{2,p}$. Note that if $\boldsymbol{XW} = \boldsymbol{Y} + \boldsymbol{E}$ is inconsistent, especially when the number of data samples m is greater than the number of features n, least-square solution space of the equation can be used as a substitute. Actually, we only need to solve $\boldsymbol{XW} = \boldsymbol{XX}^+(\boldsymbol{Y} + \boldsymbol{E})$, where $\boldsymbol{X}^+$ is pseudo-inverse. This equation is compatible when $\boldsymbol{X}$ is row full rank, since $\boldsymbol{XX}^+ = \boldsymbol{I}$ on this occasion. Gaussian Elimination is a simple and efficient way to obtain the solution space of $\boldsymbol{W}$. Without loss the generality, we assume that the first m_0 column vectors of $\boldsymbol{X}$ are linearly independent, *i.e.*

$$\begin{aligned} [\,\boldsymbol{X} \vdots \boldsymbol{XX}^+(\boldsymbol{Y}+\boldsymbol{E})\,] &= [\,\boldsymbol{X}_1 \ \boldsymbol{X}_2 \vdots \boldsymbol{XX}^+(\boldsymbol{Y}+\boldsymbol{E})\,] \\ &\xrightarrow{\text{left-multiply } \boldsymbol{D}} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{M} \vdots & \boldsymbol{N}+\boldsymbol{LE} \\ \mathbf{0} & \mathbf{0} \vdots & \mathbf{0} \end{bmatrix}, \end{aligned} \quad (7)$$

where the rank of $\boldsymbol{X}$ is m_0, $\boldsymbol{X}_1 \in \mathbb{R}^{m\times m_0}$, $\boldsymbol{X}_2 \in \mathbb{R}^{m\times n_0}$, $\boldsymbol{I} \in \mathbb{R}^{m_0\times m_0}$, $\boldsymbol{M} \in \mathbb{R}^{m_0\times n_0}$, $\boldsymbol{N} \in \mathbb{R}^{m_0\times c}$, and $\boldsymbol{D} \in \mathbb{R}^{m_0\times m_0}$ is product matrix of a series of elemental matrices, $\boldsymbol{L} = \boldsymbol{DXX}^+ \in \mathbb{R}^{m_0\times m_0}$, and $n_0 = n + 1 - m_0$. Thus, the solution space of $\boldsymbol{W}$ is

[1] If $0 < r < 1$ or $0 < p < 1$, $\ell_{r,p}$-norm does not satisfy triangle inequality. However, this does not affect the proposed algorithm.

$$\boldsymbol{W} = \mathbf{P}\boldsymbol{U} + \boldsymbol{Q} + \boldsymbol{F} = \begin{bmatrix}\boldsymbol{M}\\ \boldsymbol{I}\end{bmatrix}\boldsymbol{U} + \begin{bmatrix}\boldsymbol{N}\\ \mathbf{0}\end{bmatrix} + \begin{bmatrix}\boldsymbol{LE}\\ \mathbf{0}\end{bmatrix} \quad (8)$$

where $\boldsymbol{P} \in \mathbb{R}^{n\times n_0}$, $\boldsymbol{Q} \in \mathbb{R}^{n\times c}$, $\boldsymbol{F} \in \mathbb{R}^{n\times c}$, $\boldsymbol{I} \in \mathbb{R}^{n_0\times n_0}$, and $\boldsymbol{U} \in \mathbb{R}^{n_0\times c}$ is an arbitrary matrix.

Finally, the optimization problem of Eqn. (5) can be reformulated as

$$\min_{\boldsymbol{U},\boldsymbol{E}} \left\| \boldsymbol{PU} + \boldsymbol{Q} + \begin{bmatrix}\boldsymbol{LE}\\ \mathbf{0}\end{bmatrix} \right\|_{2,p}, \; s.t., \; \boldsymbol{Y} \odot \boldsymbol{E} \geqslant \mathbf{0} \quad (9)$$

where $0 < p \leq 1$.

3 An Iterative Algorithm for DSO-FS

We propose an iterative algorithm to solve the optimization problem of Eqn. (9) due to that no analytical solution is available. At each iteration step, we alternately optimize variables $\boldsymbol{U}$ and $\boldsymbol{E}$. An objective function with $\ell_{2,\mathrm{p}}$-norm ($0 < p \leq 1$) is non-smooth and non-convex when $0 < p < 1$. To efficiently solve this problem, $\ell_{2,\mathrm{p}}$-norm is reformulated by Frobenius-norm ($\mathcal{F}$-norm) that is smooth and convex, as

$$\|\boldsymbol{A}\|_{2,p}^{p} = \|\boldsymbol{\Sigma A}\|_F^2, \quad (10)$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix with the i-th diagonal element $\boldsymbol{\Sigma}_{ii} = 1/\|\boldsymbol{a}_i\|_2^{1-p/2}$, and $\|\boldsymbol{a}_\mathrm{i}\|_2$ is defined in Eqn.(1)[2]. Note that $\boldsymbol{Z}^* = \operatorname{argmin}_{\boldsymbol{Z}} \|\boldsymbol{A}(\boldsymbol{Z})\|_{2,p}^{p} = \operatorname{argmin}_{\boldsymbol{Z}} \|\boldsymbol{A}(\boldsymbol{Z})\|_{2,p}$, where $\boldsymbol{A}(\boldsymbol{Z})$ is a function of $\boldsymbol{Z}$. The similar strategies has also been adopted in FOcal Underdetermined System Solver (FOCUSS) (Cotter et al., 2005; Gorodnitsky & Rao, 1997), and various optimization techniques for $\ell_{2,1}$-norm regularization based feature selection algorithms (Hou et al., 2011; Krishnapuram et al., 2005; Merchante et al., 2012; Nie, et al., 2010; Xiang, et al., 2012; Yi Yang et al., 2011) can be considered as its special case when $p = 1$. Our solution is summarized in Algorithm 1.

Algorithm 1. Feature Selection via Direct Sparsity Optimization (DSO-FS)

Input: data points $\{\boldsymbol{x}^i\}_{i=1}^m (\boldsymbol{x}^i \in \mathbb{R}^n)$ and their corresponding label $\{y^i\}_{i=1}^m$; norm power p; number of features d to be selected.

Construct $\boldsymbol{X}$ and $\boldsymbol{Y}$ according to Eqn. (2), and $\boldsymbol{L}$, $\boldsymbol{M}$, $\boldsymbol{N}$, $\mathbf{P}$ and $\boldsymbol{Q}$ according to Eqn. (7) and Eqn. (8).

Set $k = 0$ and initialize $\boldsymbol{E}^0$ and $\boldsymbol{\Sigma}^0 \in \mathbb{R}^{n\times n}$ with a zero matrix and an identity matrix respectively.

repeat

$\boldsymbol{G} = \boldsymbol{Q} + \begin{bmatrix}\boldsymbol{LE}^k\\ \mathbf{0}\end{bmatrix}$

$\boldsymbol{U}^{k+1} = \operatorname{argmin}_{\boldsymbol{U}} \|\boldsymbol{\Sigma}^k(\boldsymbol{PU} + \boldsymbol{G})\|_F^2$

$\boldsymbol{V}^k = -\boldsymbol{MU}^{k+1} + \boldsymbol{N} + \boldsymbol{LE}^k$.

Update diagonal matrix $\boldsymbol{\Lambda}^k$, where the i-th diagonal element is $\frac{1}{\|\boldsymbol{v}_i^k\|_2^{1-p/2}}$

$\boldsymbol{H} = -\boldsymbol{MU}^{k+1} + \boldsymbol{N}$

$\boldsymbol{E}^{k+1} = \operatorname{argmin}_{\boldsymbol{E}} \|\boldsymbol{\Lambda}^k(\boldsymbol{LE} + \boldsymbol{H})\|_F^2$, s.t. $\boldsymbol{Y} \odot \boldsymbol{E} \geqslant \mathbf{0}$

Update $k = k + 1$, $\boldsymbol{W}^k = \mathbf{P}\boldsymbol{U}^k + \boldsymbol{Q} + \begin{bmatrix}\boldsymbol{LE}^k\\ \mathbf{0}\end{bmatrix}$

Update $\boldsymbol{\Sigma}^k$, where the i-th diagonal element is $\frac{1}{\|\boldsymbol{w}_i^k\|_2^{1-p/2}}$

until *convergence*

Sort all features according to $\|\boldsymbol{w}_i\|_2$ and select the largest d features.

Note that at each iteration step, for solving $\boldsymbol{U}^{k+1} = \operatorname{argmin}_{\boldsymbol{U}} \|\boldsymbol{\Sigma}^k(\boldsymbol{PU} + \boldsymbol{G})\|_F^2$, an analytical solution is available, *i.e.*,

$$\boldsymbol{U}^{k+1} = -(\boldsymbol{P}^t\boldsymbol{SP})^{-1}\boldsymbol{P}^t\boldsymbol{SG}. \quad (11)$$

where $\boldsymbol{S} = (\boldsymbol{\Sigma}^k)^2$.

According to Eqn. (8), Eqn. (11) can be reformulated as

$$\boldsymbol{U}^{k+1} = (\boldsymbol{M}^t\boldsymbol{S}_1\boldsymbol{M} + \boldsymbol{S}_2)^{-1}\boldsymbol{M}^t\boldsymbol{S}_1\boldsymbol{K}, \quad (12)$$

where $\boldsymbol{S}_1 \in \mathbb{R}^{m_0\times m_0}$ is a diagonal matrix, of which the diagonal elements are the first m_0 elements of $\boldsymbol{S}$, $\boldsymbol{S}_2 \in \mathbb{R}^{n_0\times n_0}$ is a diagonal matrix, of which the diagonal elements are the last n_0 elements of $\boldsymbol{S}$, and $\boldsymbol{K} = \boldsymbol{N} + \boldsymbol{LE}^k (\in \mathbb{R}^{m_0\times c})$.

If $m_0 < n_0$, the optimization problem's computational cost can be further reduced by a simple matrix operation

$$\boldsymbol{U}^{k+1} = \boldsymbol{T}(\boldsymbol{MT} + \boldsymbol{I})^{-1}\boldsymbol{K}, \quad (13)$$

where $\boldsymbol{T} = \boldsymbol{S}_2^{-1}\boldsymbol{M}^t\boldsymbol{S}_1$ and $\boldsymbol{I} \in \mathbb{R}^{m_0\times m_0}$.

For feature selection problems with the number of samples less than the number of features, i.e., $m_0 \leq m \ll n_0$, updating Eqn. (13) has reduced computational cost than Eqn. (11). It is worth noting that it is not necessary to directly calculate $\boldsymbol{C} = (\boldsymbol{MT} + \boldsymbol{I})^{-1}\boldsymbol{K}$, which is computationally more expensive than solving the following linear equation

$$(\boldsymbol{MT} + \boldsymbol{I})\boldsymbol{C} = \boldsymbol{K}. \quad (14).$$

Although no analytical solution is available for $\boldsymbol{E}^{k+1} = \operatorname{argmin}_{\boldsymbol{E}} \|\boldsymbol{\Lambda}^k(\boldsymbol{LE} + \boldsymbol{H})\|_F^2$ (s.t. $\boldsymbol{Y} \odot \boldsymbol{E} \geqslant \mathbf{0}$), it is a smooth and convex optimization problem. Therefore, it can be efficiently solved by existing tools, such as CVX (CVX Research, 2011).

4 Convergence proof

Algorithm 1 makes $\|\boldsymbol{W}\|_{2,p}$ to monotonically decrease at every iteration step and finally converges.

Lemma 1. *Given any two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, we have*

$$(1-\theta)\|\boldsymbol{a}\|_2^2 + \theta\|\boldsymbol{b}\|_2^2 \geq \|\boldsymbol{a}\|_2^{2-2\theta}\|\boldsymbol{b}\|_2^{2\theta}, \quad (15)$$

where $0 < \theta < 1$ and the equality holds if and only if $a = b$.

Proof. Since $\ln(x^2)$ is concave, we have

[2] $\boldsymbol{\Sigma} = \boldsymbol{\Pi}^{-1}$, where $\boldsymbol{\Pi}$ is a diagonal matrix with the i-th diagonal element $\boldsymbol{\Pi}_{ii} = \|\boldsymbol{a}_i\|_2^{1-p/2}$. If any row vector $\boldsymbol{a}_i = 0$, we define $\boldsymbol{\Sigma} = \boldsymbol{\Pi}^{+}$.

$$\ln\left((1-\theta)x_1^2+\theta x_2^2\right) \geq (1-\theta)\ln(x_1^2)+\theta\ln(x_2^2), \quad (16)$$

where $0<\theta<1$. The equality holds if and only if $x_1 = x_2$, indicating that

$$(1-\theta)x_1^2+\theta x_2^2 \geq x_1^{2-2\theta}x_2^{2\theta}. \quad (17)$$

Then, we have

$$(1-\theta)\|\boldsymbol{a}\|_2^2 + \theta\|\boldsymbol{b}\|_2^2 \geq \|\boldsymbol{a}\|_2^{2-2\theta}\|\boldsymbol{b}\|_2^{2\theta}, \quad (18)$$

where the equality holds if and only if $\boldsymbol{a}=\boldsymbol{b}$. □

Lemma 2. *Given an optimization problem:*

$$\min_{\boldsymbol{Z}} \| \boldsymbol{S}\boldsymbol{\Phi}(\boldsymbol{Z})\|_F^2, \ \ s.t.\ \boldsymbol{Z}\in\boldsymbol{\mathcal{F}},$$

where $\boldsymbol{\Phi}(\boldsymbol{Z})$ is a function of $\boldsymbol{Z}$, $\boldsymbol{\mathcal{F}}$ is the feasible region, and $\boldsymbol{S}$ is a diagonal matrix whose i-th diagonal element is $1/\|\boldsymbol{\Phi}(\boldsymbol{Z}_0)_i\|_2^{1-p/2}$ ($\boldsymbol{Z}_0$ could be any object in $\boldsymbol{\mathcal{F}}$, $\boldsymbol{\Phi}(\boldsymbol{Z}_0)_i$ is the i-th row vector of $\boldsymbol{\Phi}(\boldsymbol{Z}_0)$ and $0<p\leq 2$), we have

$$\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)\|_{2,p} \leq \|\boldsymbol{\Phi}(\boldsymbol{Z_0})\|_{2,p}, \quad (19)$$

where $\boldsymbol{Z}^*$ is the optimal solution of Eqn. (19) and the equality holds if and only if $\boldsymbol{\Phi}(\boldsymbol{Z}^*)=\boldsymbol{\Phi}(\boldsymbol{Z}_0)$

Proof. Since $\boldsymbol{Z}^*$ is the optimal solution, we have

$$\| \boldsymbol{S}\boldsymbol{\Phi}(\boldsymbol{Z}^*)\|_F^2 \leq \| \boldsymbol{S}\boldsymbol{\Phi}(\boldsymbol{Z_0})\|_F^2. \quad (20)$$

Then

$$\sum_i \frac{\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^2}{\|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^{2-p}} \leq \sum_i \|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^p. \quad (21)$$

where $\boldsymbol{\Phi}(\boldsymbol{Z_0})_i$ and $\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i$ are the i-th row vector of $\boldsymbol{\Phi}(\boldsymbol{Z}_0)$ and $\boldsymbol{\Phi}(\boldsymbol{Z}^*)$, respectively.

According to Lemma 1, we have

$$\left(1-\frac{p}{2}\right)\|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^2 + \frac{p}{2}\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^2 \geq \|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^{2-p}\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^p. \quad (22)$$

Then dividing the both sides by $\|\boldsymbol{\Phi}(\boldsymbol{Z}_0)_i\|_2^{2-p}$, we have

$$\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^p \leq \left(1-\frac{p}{2}\right)\|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^p + \frac{p}{2}\frac{\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^2}{\|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^{2-p}} \quad (23)$$

It indicates that

$$\sum_i \|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^p \leq \left(1-\frac{p}{2}\right)\sum_i \|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^p + \frac{p}{2}\sum_i \frac{\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^2}{\|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^{2-p}}. \quad (24)$$

Combining Eqns. (21) and (24), we obtain

$$\sum_i \|\boldsymbol{\Phi}(\boldsymbol{Z}^*)_i\|_2^p \leq \sum_i \|\boldsymbol{\Phi}(\boldsymbol{Z_0})_i\|_2^p. \quad (25)$$

Therefore,

$$\|\boldsymbol{\Phi}(\boldsymbol{Z}^*)\|_{2,p} \leq \|\boldsymbol{\Phi}(\boldsymbol{Z_0})\|_{2,p}, \quad (26)$$

where the equality holds if and only if $\boldsymbol{\Phi}(\boldsymbol{Z}^*)=\boldsymbol{\Phi}(\boldsymbol{Z_0})$. □

Theorem 1. *The sequence $\{\boldsymbol{W}^k\}$ produced via Algorithm 1 has the following properties: $\|\boldsymbol{W}^k\|_{2,p}$ is non-increasing at* successive *iteration steps and $\{\|\boldsymbol{W}^k\|_{2,p}\}$ converges to a limited value.*

Proof. Supposing we have obtained the solution $\boldsymbol{U}^k$, $\boldsymbol{E}^k$, and the objective function $\boldsymbol{W}^k$ at the $(k+1)$-*th* iteration step, we solve the optimization problem $\min_{\boldsymbol{U}}\left\|\boldsymbol{\Sigma}^k(\boldsymbol{PU}+\boldsymbol{G})\right\|_F^2$ to obtain $\boldsymbol{U}^{k+1}$ by fixing $\boldsymbol{E}^k$. According to Lemma 2, we have

$$\left\|\boldsymbol{V}^k\right\|_{2,s} = \left\| \mathbf{P}\boldsymbol{U}^{k+1} + \boldsymbol{Q} + \begin{bmatrix}\boldsymbol{LE}^k \\ \mathbf{0}\end{bmatrix}\right\|_{2,p} \leq \|\boldsymbol{W}^k\|_{2,p}. \quad (27)$$

Then we fix $\boldsymbol{U}^{k+1}$, and solve the optimization problem $\min_{\boldsymbol{E}}\left\|\boldsymbol{\Lambda}^k(\boldsymbol{LE}+\boldsymbol{H})\right\|_F^2$ to obtain $\boldsymbol{E}^{k+1}$. According to Lemma 2, we have

$$\|\boldsymbol{W}^{k+1}\|_{2,p} = \left\|\mathbf{P}\boldsymbol{U}^{k+1} + \boldsymbol{Q} + \begin{bmatrix}\boldsymbol{LE}^{k+1} \\ \mathbf{0}\end{bmatrix}\right\| \leq \|\boldsymbol{V}^k\|_{2,p}. \quad (28)$$

Combining (27) and (28), we obtain

$$\|\boldsymbol{W}^{k+1}\|_{2,p} \leq \|\boldsymbol{W}^k\|_{2,p}, \quad (29)$$

where the equality holds if and only if $\mathbf{W}^{k+1}=\mathbf{V}^k=\mathbf{W}^k$. Since the lower bound of $\|\boldsymbol{W}\|_{2,p}$ is limited, $\{\|\boldsymbol{W}^k\|_{2,p}\}$ will converge. □

Theorem 2. *If sequences $\{\boldsymbol{W}^k\}$ and $\{\boldsymbol{E}^k\}$ produced in Algorithm 1 have limit points, the limit points satisfy the Karush–Kuhn–Tucker (KKT) conditions of Eqn. (6). When $p\geq 1$, the limited points are globally optimal.*

Due to space limitations, the proof is provided in supplementary document (Peng & Fan, 2016)..

It is worth noting that Lemma 2 works for sparsity regularization based feature selection algorithms, *i.e.*,

$$\min_{\boldsymbol{Z}} \sum_i \|\boldsymbol{A}_i\boldsymbol{Z}+\boldsymbol{B}_i\|_{2,p} + \lambda\|\boldsymbol{Z}\|_{2,p}, \quad (30)$$

where λ is the regularization coefficient.

Eqn. (30) can be reformulated as

$$\min_{\boldsymbol{Z}}\|[(\boldsymbol{A}_1\boldsymbol{Z}+\boldsymbol{B}_1)^t,\ldots,(\boldsymbol{A}_i\boldsymbol{Z}+\boldsymbol{B}_i)^t,\ldots,\lambda\boldsymbol{Z}^t]\|_{2,p}. \quad (31)$$

The Robust Feature Selection (RFS) (Nie, et al., 2010; Xiang, et al., 2012) is a special case of Eqn. (30) when $i=1$ and $p=1$.

5 Experiments

5.1 Experimental datasets and settings

The proposed algorithm has been evaluated based on 6 publicly available datasets. In particular, 2 datasets were obtained from UCI, including ISOLET and SEMEION. ISOLET is a speech recognition data set with 7797 samples from 26 classes, and each sample has 617 features; SEMEION contains 1593 handwritten images from ~80 people, stretched in a rectangular box of 16×16. For these 2 datasets, the number of features is less than the number of their samples. Another 2 datasets were microarray data, including LUNG and CLL-SUB-111. In particular, LUNG consists of 203 samples, each having 3312 genes with standard deviations larger than 50 expression units (Bhattacharjee et al., 2001); CLL-SUB-111 consists of 111 samples with 11340 features (featureselection.asu.edu). Our algorithm has also been validated based on 2 image datasets, including UMIST and AR. In particular, UMIST includes face images with a resolution of 56×46 from 20 different people, and AR has 130 samples with 2400 features.

We compared our method with sparsity regularization based feature selection methods, including ℓ_1-SVM(Bradley & Mangasarian, 1998; Mangasarian, 2006) and $\ell_{2,1}$-norm based Robust Feature Selection (RFS) (Nie, et al., 2010; Xiang, et al., 2012), with an objective function $\mathcal{J}(\boldsymbol{W}) = \|\boldsymbol{XW} - \boldsymbol{Y}\|_{2,1} + \lambda\|\boldsymbol{W}\|_{2,1}$. We also compared our algorithm with well-known filter methods, including ReliefF (Kira & Rendell, 1992) and mRMR (Peng, et al., 2005).

The feature selection methods were evaluated based on their classification accuracy. Particularly, linear SVM (Chang & Lin, 2011) was adopted to build classifiers based on the selected features. The parameter C of linear SVM classifiers were tuned using a cross-validation strategy by searching a candidate set [10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}]. The regularization parameter of ℓ_1-SVM and RFS were tuned using the same cross-validation strategy by searching a candidate set [10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}].

In our experiments, we first normalized all the data to have 0 mean and unit standard deviation for each feature. 10 trials were performed on each dataset. In each trial, the samples of each dataset were randomly spitted into training and testing subsets with a ratio of 6:4. For tuning parameters, a 3-fold was used for datasets with less than 200 training samples, and an 8-fold cross-validation was used for other datasets.

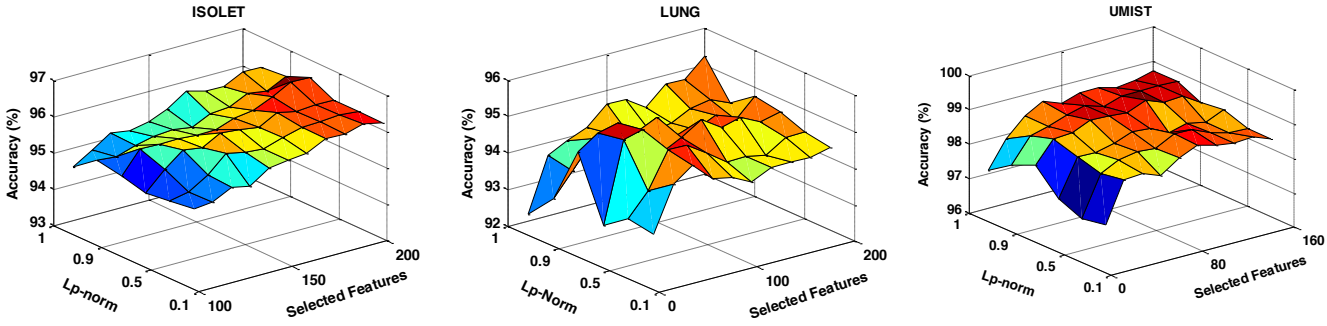
5.2 Effect of parameter p

To investigate how the classification performance is affected by the parameter p of DSO-FS, we performed experiments based on datasets ISOLET, LUNG, and UMIST. We obtained solutions of $\boldsymbol{W}$ with different settings of $p \in$ [0.1, 0.3, 0.5, 0.7, 0.9, 1] using Algorithm 1 based on the above 3 datasets, and then selected top ranked features according to ℓ_2-norm $\|\boldsymbol{w}_i\|_2$ to build classifiers. Figure 1 shows the classification accuracy as a function of the number of the selected features and the parameter p. The results shown in Figure 1 indicated that p played an important role in the classification.

5.3 Comparisons with state-of-the-art methods

Since the classification performance of linear SVM classifiers built on the features selected by our method is hinged on the parameter p, we used a cross-validation strategy to select an optimal value from [0.1, 0.3, 0.5, 0.7, 0.9, 1] for our method. Figure 2 shows the average classification performance of classifiers built on the features selected by different methods in 10 trials. In particular, the average classification accuracy is shown as a function of the number of features used in the classification model. Compared with other methods, the proposed method achieved higher classification accuracy on most datasets, indicating that our method had overall better performance than other algorithms.

Table 1 summarizes mean and standard deviation of the classification rates in 10 trails for classifiers built on the top 100 features selected by the algorithms under comparison. These results demonstrated that our algorithm had the overall best classification accuracy on all the 6 datasets.



(a) (b) (c)

Figure 1: Classification accuracy with different numbers of features selected with different values of p. The results shown were obtained based on datasets: (a) ISOLET, (b) LUNG, and (c) UMIST.

Table1: Mean and standard deviation of the classification Accuracy (%) of Linear-SVM built on the selected top 100features by different algorithms for datasets: ISOLET, SEMEION, LUNG, CLL-SUB-111, UMIST, and AR.

	mRMR	ReliefF	L1-SVM	RFS	DSO-FS
ISOLET	90.82±0.72	89.13±0.53	93.90±0.53	93.72±0.55	**95.23±0.44**
SEMEION	88.18±1.01	86.54±0.89	90.36±0.33	90.45±1.00	**90.64±0.93**
LUNG	94.88±1.20	94.88±1.42	95.12±1.22	**95.37±1.42**	**95.37±1.42**
CLL-SUB-111	76.47±11.0	72.94±9.18	79.41±4.15	75.88±6.80	**81.18±6.60**
UMIST	96.87±0.84	98.17±0.92	98.17±0.80	98.04±0.73	**98.39±1.16**
AR	87.31±5.02	85.58±6.15	87.12±9.26	88.85±6.76	**89.62±3.74**

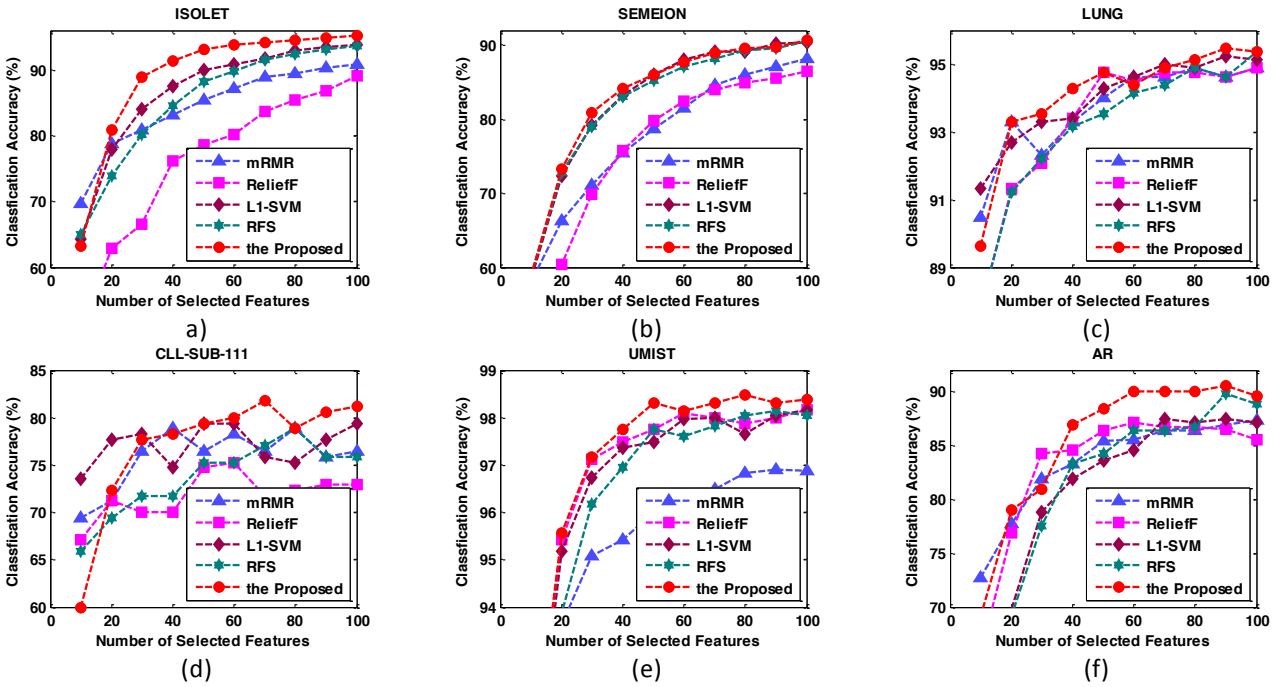


Figure 2: Average classification accuracy of 10 trials for classifiers built on the selected top 100 features by different algorithms. The results shown were obtained based on datasets: (a) ISOLET, (b) SEMEION, (c) LUNG , (d) CLL-SUB-111, (e) UMIST, and (f) AR.

6 Conclusions and Discussions

In this paper, a novel feature selection algorithm via direct sparsity optimization was proposed. Our method directly optimizes a large margin linear classification model's sparsity with $\ell_{2,p}$-norm $(0 < p \leq 1)$ subject to a data-fitting constraint, different from the sparse regularization based algorithms that typically take a trade-off between a data-fitting loss function term and a sparsity term. Since little is known about such a trade-off's impact on the feature selection, empirical parameter tuning is typically adopted for choosing the trade-off parameter. Our feature selection method adopts data-fitting inequality constraints to obtain large between-class margins, which could improve the generalization ability of the selected features, rather than data-fitting loss functions built upon the regression of class labels (Bartlett, et al., 2004). We also proposed an efficient algorithm to solve the non-convex $(0 < p < 1)$ and non-smooth optimization problem associated with the feature selection problem. Extensive experiments based on 6 datasets have demonstrated that the proposed method could achieve better performance than state-of-the-art feature selection algorithms. Furthermore, our algorithm can be easily extended for solving other sparsity regularization algorithms. In particular, our algorithm could be used to solve ℓ_0 and $\ell_{2,0}$ based optimization problems subject to linear constraints by setting p close to 0. Furthermore, similar to SVM, we could also add soft-margin to our algorithm for better performance in the future work.

Acknowledgments

This work was supported in part by National Key Basic Research and Development Program of China (No. 2015CB856404), NSFC (No. 81271514, 61473296), and NIH grants MH070365 and AG014971.

References

[Bartlett, P.L., et al., 2004] Large margin classifiers: convex loss, low noise, and convergence rates. Advances in Neural Information Processing Systems 16. 16: 1173-1180, 2004.

[Bhattacharjee, A., et al., 2001] Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. Proceedings of the National Academy of Sciences of the United States of America. 98: 13790-13795, 2001.

[Bradley, P.S. and Mangasarian, O.L., 1998] Feature selection via concave minimization and support vector machines. Proceedings of the International Conference on Machine Learning: 82-90, 1998.

[Candes, E.J. and Tao, T., 2005] Decoding by linear programming. Ieee Transactions on Information Theory. 51: 4203-4215, 2005.

[Cawley, G.C., et al., 2006] Sparse multinomial ogistic regression via Bayesian l1 regularisation. Advances inNeural Information Processing Systems: 209-216, 2006.

[Chang, C.C. and Lin, C.J., 2011] LIBSVM: A Library for Support Vector Machines. Acm Transactions on Intelligent Systems and Technology. 2, 2011.

[Chartrand, R. and Staneva, V., 2008] Restricted isometry properties and nonconvex compressive sensing. Inverse Problems. 24, 2008.

[Cotter, S.F., et al., 2005] Sparse solutions to linear inverse problems with multiple measurement vectors. Ieee Transactions on Signal Processing. 53: 2477-2488, 2005.

Cvx Research, I. CVX: Matlab software for disciplined convex programming, version 2.0, http://cvxr.com/cvx, 2011.

[Gorodnitsky, I.F. and Rao, B.D., 1997] Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm. Ieee Transactions on Signal Processing. 45: 600-616, 1997.

[Guyon, I. and Elisseeff, A., 2003] An introduction to variable and feature selection. Journal of Machine Learning Research. 3: 1157-1182, 2003.

[Guyon, I., et al., 2002] Gene selection for cancer classification using support vector machines. Machine Learning. 46: 389-422, 2002.

[Hou, C., et al., 2011] Feature Selection via Joint Embedding Learning and Sparse Regression. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, 2011.

[Kira, K. and Rendell, L.A., 1992] A Practical Approach to Feature-Selection. Machine Learning: 249-256, 1992.

[Kong, D.G. and Ding, C., 2013] Efficient Algorithms for Selecting Features with Arbitrary Group Constraints via Group Lasso. 2013 Ieee 13th International Conference on Data Mining (Icdm): 379-388, 2013.

[Kong, D.G., et al., 2014] Exclusive Feature Learning on Arbitrary Structures via L1,2-norm. Advances in Neural Information Processing Systems, 2014.

[Krishnapuram, B., et al., 2005] Sparse multinomial logistic regression: Fast algorithms and generalization bounds. Ieee Transactions on Pattern Analysis and Machine Intelligence. 27: 957-968, 2005.

[Lewis, D.D., 1992] Feature-Selection and Feature-Extraction for Text Categorization. Speech and Natural Language: 212-217, 1992.

[Liu, H., et al., 2009] Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. International Conference on Machine Learning, 2009.

[Liu, J., et al., 2009] Multi-Task Feature Learning Via Efficient L2,1-Norm Minimization. Uncertainty in Artificial Intelligence, 2009.

[Liu, Y.F., et al., 2007] Support vector machines with adaptive L-q penalty. Computational Statistics & Data Analysis. 51: 6380-6394, 2007.

[Mangasarian, O.L., 2006] Exact 1-norm support vector machines via unconstrained convex differentiable minimization. Journal of Machine Learning Research: 1517-1530, 2006.

[Merchante, L.F.S., et al., 2012] An Effcient Approach to Sparse Linear Discriminant Analysis. International Conference on Machine Learning, 2012.

[Nie, F., et al., 2010] Efficient and Robust Feature Selection via Joint L2,1-Norms Minimization. Advances in Neural Information Processing Systems, 2010.

[Obozinski, G., et al., 2006] Multi-task feature selection. Technical report, Department of Statistics, University of California, Berkeley, 2006.

[Peng, H. and Fan, Y., 2016] Supplementary material of "Direct Sparsity Optimization Based Feature Selection for Multi-Class Classification", 2016.

[Peng, H.C., et al., 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, 2005.

[Tibshirani, R., 1996] Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society Series B-Methodological. 58: 267-288, 1996.

[Wang, L., et al., 2008] Hybrid huberized support vector machines for microarray classification and gene selection. Bioinformatics. 24: 412-419, 2008.

[Xiang, S.M., et al., 2012] Discriminative Least Squares Regression for Multiclass Classification and Feature Selection. Ieee Transactions on Neural Networks and Learning Systems. 23: 1738-1754, 2012.

[Yi Yang, et al., 2011] L2,1-Norm Regularized Discriminative Feature Selection for Unsupervised Learning. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, 2011.

[Zhang, M., et al., 2014] Feature Selection at the Discrete Limit. Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014.