A Nonconvex Penalty Function with Integral Convolution Approximation for Compressed Sensing

Feng Zhang[1], Jianjun Wang[1]*, Wendong Wang[1], Jianwen Huang[1], Changan Yuan[2]

[1] *School of Mathematic & Statistics, Southwest University, Chongqing 400715, China*

[2] *Guangxi Teachers Education University, Nanning 530023, China*

Abstract. In this paper, we propose a novel nonconvex penalty function for compressed sensing using integral convolution approximation. It is well known that an unconstrained optimization criterion based on ℓ_1-norm easily underestimates the large component in signal recovery. Moreover, most methods either perform well only under the measurement matrix satisfied restricted isometry property (RIP) or the highly coherent measurement matrix, which both can not be established at the same time. We introduce a new solver to address both of these concerns by adopting a frame of the difference between two convex functions with integral convolution approximation. What's more, to better boost the recovery performance, a weighted version of it is also provided. Experimental results suggest the effectiveness and robustness of our methods through several signal reconstruction examples in term of success rate and signal-to-noise ratio (SNR).

Key words. Compressed sensing; Penalty function; Integral convolution; Restricted isometry property; Coherence

1 Introduction

The following lasso problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{ P(\boldsymbol{x};\lambda) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1 \right\} \tag{1.1}$$

with ℓ_2-norm data fidelity and ℓ_1-norm penalty, where λ is a positive parameter controlling the trade-off between the two terms, has been the subject of extensive research in various fields of science and engineering, including compressed sensing (CS) [7], regression analysis [23], signal processing [15], etc. In term of regression analysis, our goal is to approximate the response variables $\boldsymbol{y} = (y_1, \cdots, y_m)$ by exploiting the properties of lasso, i.e., the shrinkage and selection of variables. For compressed sensing, based on the hypothesis that signals can be of sparse representation, we try to reconstruct the n-dimensional original signal $\boldsymbol{x}$ from m-dimensional noisy

*Corresponding author, E-mail: wjjmath@gmail.com, wjj@swu.edu.cn(J.J. Wang) observation $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$ with noise level $\|\boldsymbol{e}\|_2 \leq \epsilon$, where A is an $m \times n$ measurement matrix with $m \ll n$ and satisfies the following *Restricted Isometry Property* (RIP) [3]:

Definition 1.1. *For an $m \times n$ measurement matrix A, the k-restricted isometry constant $0 < \delta_k < 1$ of A is the smallest quantity such that*

$$(1 - \delta_k)\|\boldsymbol{x}\|_2^2 \leq \|A\boldsymbol{x}\|_2^2 \leq (1 + \delta_k)\|\boldsymbol{x}\|_2^2$$

holds for all k-sparse signals $\boldsymbol{x}$.

Given a deterministic matrix A, it is generally NP-hard, however, to verify whether A is a RIP matrix. Fortunately, some random matrices have been proved to satisfy RIP with overwhelmingly high probability, such as Gaussian random matrices, Bernoulli random matrices and partial Fourier random matrices, etc.

ℓ_1-norm penalty is widely used as a regularizer, since among convex relaxation for ℓ_0-norm (counting the number of nonzero elements) it has a more concise form. However, the obtained solution by this method is not necessarily the most sparse solution and minimizing ℓ_1 problem can easily lead to underestimate high-amplitude components compared with ℓ_0 regularised optimisation, which is illustrated by threshold functions of ℓ_0 problem and ℓ_1 problem in Fig.1.1.

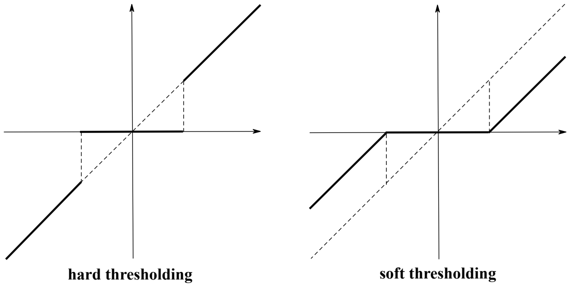


Fig. 1.1: Thresholding function

For the limit of ℓ_1-norm, many known nonconvex surrogates of ℓ_1-norm have been proposed, such as ℓ_p-norm $(0 < p \leq 1)$ [25, 27], Exponential-Type Penalty (ETP) [9], Fraction Function Penalty [14], Smoothly Clipped Absolute Deviation (SCAD) [8] and ℓ_1-ℓ_2 [26, 30]. Studies have shown that the nonconvex penalty usually induces sparsity more effective than convex penalty in signal recovery. It is noteworthy that ℓ_1-ℓ_2 outperforms many state-of-the-art methods when A has high coherence. To be formal, one defines the *coherence* of a matrix A as

$$\mu(A) = \max_{i \neq j} \frac{|\langle A_i, A_j \rangle|}{\|A_i\|_2 \|A_j\|_2},$$

where A_i and A_j denote the i-th and j-th columns of A. We say that a matrix is high coherence if μ is big. Specifically, a RIP matrix has small coherence in general. Yet regrettably, from numerical experiments of [30], ℓ_1-ℓ_2 method suffers from a poor performance when A is a RIP matrix. Inspired by the above conclusion, we propose the following novel criterion which addresses this concern

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \big\{ J(\boldsymbol{x}; \lambda) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 + \lambda \Phi(\boldsymbol{x}; \theta, q) \big\}, \tag{1.2}$$

where the parameterized function $\Phi(\boldsymbol{x};\theta,q)$ is defined as the difference between two convex functions denoted by $\Phi(\boldsymbol{x};\theta,q)=f_1(\boldsymbol{x})-f_2(\boldsymbol{x};\theta,q)$. Concretely, in this paper, ℓ_1-norm is adopted as f_1 and f_2 is defined as the convolution of generalized Gauss function [16] with the absolute value function. There is a notably different point from ℓ_1 penalty. $\Phi(\boldsymbol{x};\theta,q)$ does not penalize large values of signals, thus which will not cause that underestimating high-amplitude components. Although our penalty function is concave, the parameterized $\Phi(\boldsymbol{x};\theta,q)$ allows us to more flexibly analyze the convexity of the objective function $J(\boldsymbol{x};\lambda)$. The question remains of how to solve (1.2) effectively. The difference of convex functions algorithm (DCA) is a descent method without line search introduced by Tao and An [20, 21]. Owing to the objective function $J(\boldsymbol{x};\lambda)$ can be considered as a difference between two parts of $\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2+f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x};\theta,q)$, thus we use the DCA to find the optimal value of (1.2) iteratively in the alternating direction method of multipliers (ADMM) framework [2]. Numerical experiments in Section 5 show that our method can reduce the requirements for sampling number and possess a good performance whether a RIP matrix or a high coherence matrix.

Candès and Wakin [4] have pointed out that: larger coefficients are penalized more heavily in the ℓ_1-norm than smaller coefficients, unlike the more democratic penalization of the ℓ_0-norm. To address this imbalance, they propose a weighted formulation of ℓ_1 minimization designed to more democratically penalize nonzero coefficients. Just like its "unweighted" counterpart (1.1), the "weighted" ℓ_1 minimization problem can be expressed as

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\left\{P_w(\boldsymbol{x};\lambda)=\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2+\lambda\|W\boldsymbol{x}\|_1\right\}, \tag{1.3}$$

where W is the diagonal matrix with positive weights $w_1,\cdots,w_n$ on the diagonal and zeros elsewhere. There is a range of valid weights for will help us to find the more sparse and accurate solution. An example of 2 dimensions is illustrated in Fig.1.2. In order to further improve the performance of our model for signal recovery, we present a weighted version of (1.2) as follows

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\left\{J_w(\boldsymbol{x};\lambda)=\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2+\lambda\Phi_w(\boldsymbol{x};\theta,q)\right\}. \tag{1.4}$$

In Algorithm 5.2, a valid set of weights can be obtained.

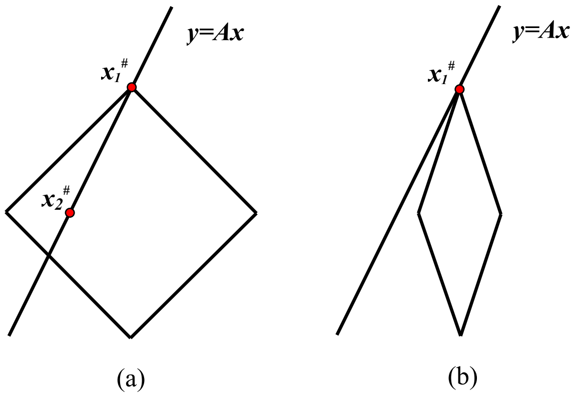


Fig. 1.2: (a) "unweighted" ℓ_1 ball. There exists $\boldsymbol{x}_2^\sharp\neq\boldsymbol{x}_1^\sharp$ such that $\|\boldsymbol{x}_2^\sharp\|_1\leq\|\boldsymbol{x}_1^\sharp\|_1$. (b) "weighted" ℓ_1 ball. There exists no $\boldsymbol{x}_2^\sharp\neq\boldsymbol{x}_1^\sharp$ such that $\|W\boldsymbol{x}_2^\sharp\|_1\leq\|W\boldsymbol{x}_1^\sharp\|_1$.

The paper is organized as follows. In Section 2, we introduce some notations and definitions in term of convolution. In Section 3, we construct a concave penalty function and analyze the convexity of the objective function. In Section 4, we extend the results of the Section 3 to the multivariate case. In Section 5, this optimization problem is done by DCA based on ADMM and we carry out some numerical simulation experiments on signal reconstruction and recovery performance of several CS solvers is present. Finally, the conclusion is addressed in Section 6.

2 Preliminaries

We commence with a recall of the relevant background material.

2.1 Notations

We use lower case letters for the entries, e.g. x_j, bold lower case letters for vectors, e.g. $\boldsymbol{x}$ and upper case letters for matrices, e.g. X. For any vector $\boldsymbol{x} \in \mathbb{R}^n$, its ℓ_p-norm $(0 < p < \infty)$ is defined $\|\boldsymbol{x}\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$. $\langle \cdot, \cdot \rangle$, $(\cdot)^T$ and $\exp(\cdot)$ stand for the inner product, transpose and exponential function, respectively. $df(\cdot)$ and $\nabla f(\cdot)$ stand for the differential and gradient of the function f. I_n is the identity matrix of dimension n. $A - B \succcurlyeq 0$, that is $A \succcurlyeq B$, means that $A - B$ is positive semidefinite. $\boldsymbol{x}^{(l)}$ represents the value of the l-th iteration of $\boldsymbol{x}$. The diagonal matrix $\mathrm{Diag}\{x_j, j = 1, \cdots, n\}$ has x_j as its j-th diagonal entry.

2.2 Two existing convolution techniques

Definition 2.1. *The infimal convolution (or epi-sum) [1] of two functions η and ξ is defined as*

$$(\eta \# \xi)(\boldsymbol{x}) := \inf_{\boldsymbol{c} \in \mathbb{R}^n} \{\eta(\boldsymbol{c}) + \xi(\boldsymbol{x} - \boldsymbol{c})\}.$$

In the notation of infimal convolution, the Moreau envelope [1] with a scale parameter $\theta > 0$ of function η is defined as

$$e\eta(\boldsymbol{x}; \theta) = \{\eta \# \frac{1}{2\theta} \| \cdot \|_2^2\}(\boldsymbol{x}) = \inf_{\boldsymbol{c} \in \mathbb{R}^n} \{\eta(\boldsymbol{c}) + \frac{1}{2\theta} \|\boldsymbol{x} - \boldsymbol{c}\|_2^2\}.$$

Another approach is integral convolution [19]. Before introducing its definition, we first present the concept of (non-negative) mollifier [6] as follows:

Definition 2.2. *If η is a smooth function on $\mathbb{R}^n$, satisfying the following four requirements*

(a) it has a bounded support;

(b) $\int_{\mathbb{R}^n} \eta(\boldsymbol{c}) d\boldsymbol{c} = 1$;

(c) $\lim\limits_{\theta \to 0} \eta(\boldsymbol{c}; \theta) = \lim\limits_{\theta \to 0} \theta^{-n} \eta(\boldsymbol{c}/\theta) = \delta(\boldsymbol{c})$, where $\delta(\boldsymbol{c})$ is the Dirac delta function and the limit must be understood in the space of Schwartz distributions, then η is a mollifier;

(d) $\eta(\boldsymbol{c}) \geq 0$, then it is called a non-negative mollifier.

The integral convolution of two functions is defined as:

Definition 2.3. *Let $\eta(\cdot;\theta)$ is a mollifier on $\mathbb{R}^n$ and $\xi(\cdot)$ is a locally integrable function written as $\xi(\cdot) \in L^1_{loc}(\Omega)$, where $\Omega \subseteq \mathbb{R}^n$. We extend $\xi(\cdot)$ to all $\mathbb{R}^n$ by setting it to be equal to zero outside Ω. The convolution of $\eta(\cdot;\theta)$ and $\xi(\cdot)$ is written $\eta(\cdot;\theta) * \xi(\cdot)$, which is defined by*

$$\{\eta(\cdot;\theta) * \xi\}(\boldsymbol{x}) := \int_{\mathbb{R}^n} \eta(\boldsymbol{x} - \boldsymbol{c};\theta)\xi(\boldsymbol{c})d\boldsymbol{c} = \int_{\mathbb{R}^n} \eta(\boldsymbol{c};\theta)\xi(\boldsymbol{x} - \boldsymbol{c})d\boldsymbol{c}$$

converges uniformly to $\xi(\cdot)$ on Ω, as $\theta \to 0$.

3 Penalty Function in Scalar Case

In this section, we present a novel penalty for the case of a scalar scope via the integral convolution technique, which promotes the sparsity of solution to a great extent. It is formed in the way of contrasting with the infimal convolution, the process of which consists of function approximate and doing subtraction of two convex functions. Further, a convex condition is proposed to ensure that the scalar objective function is convex.

3.1 Improving sparsity

The objective function of (1.3) can be re-expressed as

$$P_w(\boldsymbol{x};\lambda) = \frac{1}{2}\sum_{i=1}^{m}(y_i - \sum_{j=1}^{n} a_{ij}x_j)^2 + \lambda\sum_{j=1}^{n}|w_j x_j|.$$

Let's first consider a single element setting, then the above formula takes the form

$$P_w(x_j;\lambda) = \frac{1}{2}(y_i - a_{ij}x_j)^2 + \lambda|w_j x_j|. \tag{3.5}$$

Notice that there is a non-smooth part in (3.5), namely the absolute value function. This is not a good property for algorithmic design. It is natural to ask: How can one approximate the absolute value function with a smooth function? The most famous and very useful answer to this question is provided by the Moreau envelope. For the absolute value function, the Huber function [12] is a fairly normative instance of the Moreau envelope. The details are as follows:

Definition 3.1. *The Huber approximation (HA) of absolute value function on $\mathbb{R}$ is defined as*

$$h(x_j;\theta) := \begin{cases} \dfrac{x_j^2}{2\theta}, & |x_j| \leq \theta; \\ |x_j| - \dfrac{\theta}{2}, & |x_j| \geq \theta. \end{cases}$$

In the framework of infimal convolution, the Huber function can be written equivalently as

$$h(x_j;\theta) = \{|\cdot|\#\frac{1}{2\theta}(\cdot)^2\}(x_j) = \min_{c\in\mathbb{R}}\{|c| + \frac{1}{2\theta}(x_j - c)^2\}. \tag{3.6}$$

Meanwhile, (3.6) also means that infimal convolution is exact, i.e., the infimum is acquired for certain c. If we consider the weighted technique mentioned in the introduction, then its definition is given by:

Definition 3.2. *The weighted-based Huber approximation (WHA) of absolute value function on $\mathbb{R}$ is defined as*

$$h_w(x_j;\theta) := \begin{cases} \dfrac{(w_jx_j)^2}{2\theta}, & |w_jx_j| \leq \theta; \\ |w_jx_j| - \dfrac{\theta}{2}, & |w_jx_j| \geq \theta, \end{cases}$$

and

$$h_w(x_j;\theta) = \{|\cdot|\#\frac{1}{2\theta}(\cdot)^2\}(w_jx_j) = \min_{c\in\mathbb{R}}\{|w_jc| + \frac{1}{2\theta}(w_jx_j - w_jc)^2\}.$$

When it comes to this, we have to ask whether there is a better smooth approximation to the absolute value function. It is no doubt that the answer is affirmative. Other approaches to smoothing are Ghomi's integral convolution method [10], Seeger's ball rolling technique [17], and Teboulle's entropic proximal mappings [22]. What we are interested in is such a smooth function computed via integral convolution with a generalized Gaussian function represented in the form:

Definition 3.3. *The generalized Gaussian function (GGF) is a parametric family of continuous function on $\mathbb{R}$ defined by*

$$g(x_j;\mu,\theta,q) := \frac{q}{2\theta\Gamma(1/q)}\exp\{-(\frac{|x_j-\mu|}{\theta})^q\},$$

where $\mu, \theta > 0, q > 0$ are mean, scale parameter and shape parameter, respectively. Γ denotes the gamma function by $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$.

Remark 3.4. *The GGF includes all normal and Laplace distributions. To speak in detail, the GGF reduces to the Laplace distribution when $q = 1$; the GGF degenerates into the normal distribution when $q = 2$; moreover, if $q \to 0$, then the limit distribution of GGF is the Dirac delta function; and if $q \to \infty$, then the uniform distribution will be as the limit of GGF. These statements are illustrated in Fig.3.3.*

Since our goal is to approximate absolute value function (Y-axis symmetry) by jointly Mollifiers and the operation of convolution, we then analyze whether $g(x_j;\theta,q)$ satisfies four conditions in the definition 2.2 in the case $\mu = 0$. First of all, $g(x_j;\theta,q)$ is first order derivable at $x = 0$ provided that $q > 1$. In addition, though bounded support is not available for $g(x_j;\theta,q)$, the function is coercive so that this condition is approximately met. Moreover, we have

$$\begin{aligned}\int_{-\infty}^{\infty} g(x_j;\theta,q)dx_j &= \int_{-\infty}^{\infty}\frac{q}{2\theta\Gamma(1/q)}\exp\{-(\frac{|x_j|}{\theta})^q\}dx_j = \frac{q}{\theta\Gamma(1/q)}\int_0^\infty \exp\{-(\frac{x_j}{\theta})^q\}dx_j \\ &= \frac{1}{\Gamma(1/q)}\int_0^\infty z^{\frac{1}{q}-1}e^{-z}dz = \frac{1}{\Gamma(1/q)}\Gamma(1/q) = 1.\end{aligned}$$

Furthermore, for fix q, it's easy to check that when $x_j = 0$, $\lim\limits_{\theta\to 0} g(0;\theta,q) \to \infty$ and when $x_j \neq 0$, we have

$$\lim_{\theta\to 0} g(x_j;\theta,q) = \lim_{\theta\to 0}\frac{q}{2\theta\Gamma(1/q)}\exp\{-(\frac{|x_j|}{\theta})^q\} = \frac{q}{2\Gamma(1/q)}\lim_{\theta\to 0}\frac{1}{\theta\exp\{(\frac{|x_j|}{\theta})^q\}} \overset{t=\frac{1}{\theta}}{=} \frac{q}{2\Gamma(1/q)}\lim_{t\to\infty}\frac{t}{\exp\{(|x_j|t)^q\}},$$

if $q \geq 1$,

$$\frac{q}{2\Gamma(1/q)}\lim_{t\to\infty}\frac{t}{\exp\{(|x_j|t)^q\}} \overset{(1)}{=} \frac{1}{2\Gamma(1/q)}\lim_{t\to\infty}\frac{1}{|x_j|(|x_j|t)^{q-1}\exp\{(|x_j|t)^q\}} = 0;$$

and if $\frac{1}{r} \leq q < 1$ $(r \neq 1)$,

$$\frac{q}{2\Gamma(1/q)} \lim_{t\to\infty} \frac{t}{\exp\{(|x_j|t)^q\}} \overset{(r)}{=} \frac{(1-q)(1-2q)\cdots(1-rq+q)}{2q^{r-1}\Gamma(1/q)} \lim_{t\to\infty} \frac{1}{|x_j|(|x_j|t)^{rq-1}\exp\{(|x_j|t)^q\}} = 0;$$

further if $\frac{1}{r+1} \leq q < \frac{1}{r}$,

$$\frac{q}{2\Gamma(1/q)} \lim_{t\to\infty} \frac{t}{\exp\{(|x_j|t)^q\}} \overset{(r+1)}{=} \frac{(1-q)(1-2q)\cdots(1-rq)}{2q^{r}\Gamma(1/q)} \lim_{t\to\infty} \frac{1}{|x_j|(|x_j|t)^{rq+q-1}\exp\{(|x_j|t)^q\}} = 0,$$

where $\overset{(1)}{=}$, $\overset{(r)}{=}$, and $\overset{(r+1)}{=}$ mean that utilizing one, r, and $r+1$ times L'Hôpital's rule, respectively. So, these just verifies the (c) of definition 2.2. Last, it is obvious that $g(x_j;\theta,q) \geq 0$ for $\theta > 0, q > 0$. In conclusion, $g(x_j;\theta,q)$ is an approximate non-negative mollifier.

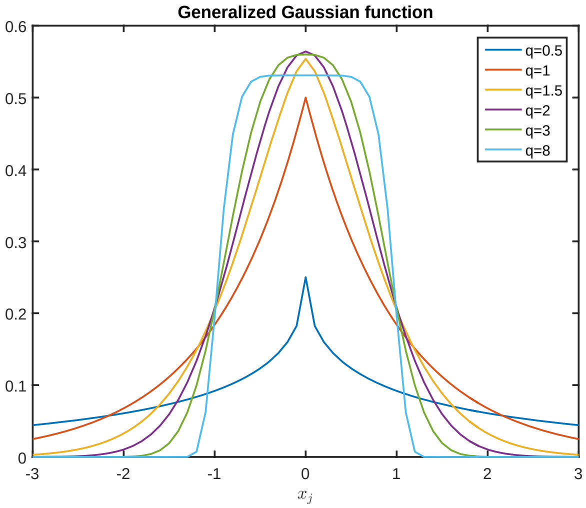


Fig. 3.3: The generalized Gaussian function (GGF) with $\mu = 0$, $\theta = 1$ and different values of q.

Based on the above analysis and inspired by definition 2.3, we have the following results

$$\psi(x_j;\theta,q) = \{g(\cdot;\theta,q) * |\cdot|\}(x_j) = \frac{q}{2\theta\Gamma(1/q)} \int_{-\infty}^{\infty} |x_j - t| \exp\{-(\frac{|t|}{\theta})^q\} dt,$$

which is an approximate to the absolute value function $|x_j|$ and called the integral convolution approximate (ICA). Similarly, the weighted-based convolution approximate (WICA) is defined as

$$\psi_w(x_j;\theta,q) = \{g(\cdot;\theta,q) * |\cdot|\}(w_j x_j) = \frac{q}{2\theta\Gamma(1/q)} \int_{-\infty}^{\infty} |w_j x_j - t| \exp\{-(\frac{|t|}{\theta})^q\} dt.$$

By some simple calculations, one can verify the following results:

Proposition 3.5. *For any $\theta > 0$, $q \in \mathbb{E}_+$ (positive even numbers) and $x_j \in \mathbb{R}$, $\psi_w(x_j;\theta,q)$ can be expressed as*

$$\psi_w(x_j;\theta,q) = w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}) + \frac{\theta\Gamma(2/q)}{\Gamma(1/q)}\{1 - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\}, \tag{3.7}$$

where

$$\mathrm{erf}_q(x) = \frac{q}{\Gamma(1/q)} \int_0^x \exp(-t^q) dt \ \forall x \in \mathbb{R},$$

which is referred to as the generalized error function (GEF) [29].

Remark 3.6. *when $q = 2$, the GGF and GEF degenerate to Gaussian function and error function, respectively. Our efforts contain not only the existing results that are described in Lemma 2.4 of [24] but also have preferable modeling ability of data and generalization performance due to the adjustable shape parameter q. By analyzing the Fig.3.4 about the generalized error function, one can obtain several important qualities: (1) $\mathrm{erf}_q(x)$ is odd and strictly increasing on $\mathbb{R}$; (2) $\mathrm{erf}_q(x)$ is convex on $(-\infty, o]$ and concave on $(0, \infty)$; (3) $\mathrm{erf}_q(0) = 0$ and $\lim\limits_{x\to\pm\infty} erf_q(x) = \pm 1$.*

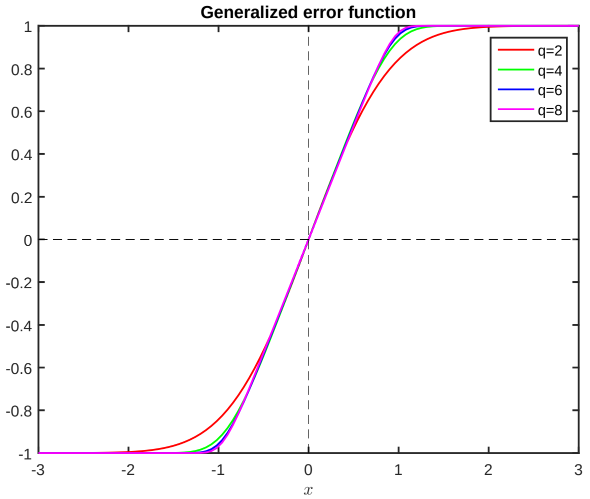


Fig. 3.4: The generalized Gaussian function (GEF) with different values of q.

Proof. Given $\theta, q, T > 0$ and let

$$L_w(x_j; \theta, q) = \frac{q}{2\theta\Gamma(1/q)} \int_{-T}^{T} |w_j x_j - t| \exp\{-(\frac{|t|}{\theta})^q\} dt,$$

then when $x_j \geq 0$, we have

$$\frac{2\theta\Gamma(1/q)}{q} L_w(x_j; \theta, q) = \int_{-T}^{T} |w_j x_j - t| \exp\{-(\frac{|t|}{\theta})^q\} dt$$

$$= \underbrace{\int_{-T}^{0} (w_j x_j - t) \exp\{-(-\frac{t}{\theta})^q\} dt}_{\triangle 1} + \underbrace{\int_{0}^{w_j x_j} (w_j x_j - t) \exp\{-(\frac{t}{\theta})^q\} dt}_{\triangle 2} + \underbrace{\int_{w_j x_j}^{T} (t - w_j x_j) \exp\{-(\frac{t}{\theta})^q\} dt}_{\triangle 3}.$$

Further calculation, we obtain

$$\begin{aligned} \triangle 1 \quad &= \quad w_j x_j \int_{-T}^{0} \exp\{-(-\frac{t}{\theta})^q\} dt - \int_{-T}^{0} t \exp\{-(-\frac{t}{\theta})^q\} dt \\ &\overset{-\frac{t}{\theta} = s}{=} \quad -w_j x_j \theta \int_{\frac{T}{\theta}}^{0} \exp(-s^q) ds - \theta^2 \int_{\frac{T}{\theta}}^{0} s \exp(-s^q) ds \\ &= \quad \frac{w_j x_j \theta \Gamma(1/q)}{q} \mathrm{erf}_q(\frac{T}{\theta}) + \frac{\theta^2 \Gamma(2/q)}{q} \mathrm{erf}_{\frac{q}{2}}(\frac{T^2}{\theta^2}), \end{aligned}$$

and

$$\begin{aligned}\triangle 2 &= w_j x_j \int_0^{w_j x_j} \exp\{-(\frac{t}{\theta})^q\}dt - \int_0^{w_j x_j} t\exp\{-(\frac{t}{\theta})^q\}dt\\ &\overset{\frac{t}{\theta}=s}{=} w_j x_j \theta \int_0^{\frac{w_j x_j}{\theta}} \exp(-s^q)ds - \theta^2 \int_0^{\frac{w_j x_j}{\theta}} s\exp(-s^q)ds\\ &= \frac{w_j x_j \theta\Gamma(1/q)}{q}\mathrm{erf}_q(\frac{w_j x_j}{\theta}) - \frac{\theta^2\Gamma(2/q)}{q}\mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2}),\end{aligned}$$

moreover

$$\begin{aligned}\triangle 3 &= \int_{w_j x_j}^{T} t\exp\{-(\frac{t}{\theta})^q\}dt - w_j x_j \int_{w_j x_j}^{T} \exp\{-(\frac{t}{\theta})^q\}dt\\ &\overset{\frac{t}{\theta}=s}{=} \theta^2 \int_{\frac{w_j x_j}{\theta}}^{\frac{T}{\theta}} s\exp(-s^q)ds - w_j x_j \int_{\frac{w_j x_j}{\theta}}^{\frac{T}{\theta}} \exp(-s^q)ds\\ &= \frac{\theta^2\Gamma(2/q)}{q}\{\mathrm{erf}_{\frac{q}{2}}(\frac{T^2}{\theta^2}) - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\} - \frac{w_j x_j \theta\Gamma(1/q)}{q}\{\mathrm{erf}_q(\frac{T}{\theta}) - \mathrm{erf}_q(\frac{w_j x_j}{\theta})\}.\end{aligned}$$

Using the fact that $\lim\limits_{x\to\infty} \mathrm{erf}_q(x) = 1$, we get

$$\begin{aligned}\psi_w(x_j;\theta,q) &= \lim_{T\to\infty} L_w(x_j;\theta,q) = \lim_{T\to\infty} \frac{(\triangle 1 + \triangle 2 + \triangle 3)q}{2\theta\Gamma(1/q)}\\ &= w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}) + \frac{\theta\Gamma(2/q)}{\Gamma(1/q)}\{1 - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\}.\end{aligned}$$

When $x_j < 0$, similar computations yield

$$\psi_w(x_j;\theta,q) = -w_j x_j \cdot \mathrm{erf}_q(-\frac{w_j x_j}{\theta}) + \frac{\theta\Gamma(2/q)}{\Gamma(1/q)}\{1 - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\}.$$

It follows from the definition of generalized error function that, $\mathrm{erf}_q(x)$ is odd on $\mathbb{R}$, i.e., $\mathrm{erf}_q(-x) = -\mathrm{erf}_q(x)$ if $q \in \mathbb{E}_+$. Therefore, for $x_j \in \mathbb{R}$ we get

$$\psi_w(x_j;\theta,q) = w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}) + \frac{\theta\Gamma(2/q)}{\Gamma(1/q)}\{1 - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\}.$$

To sum up, the proof of proposition is completed. □

Notice that the value of the absolute value function at the origin is equal to zero. Although our approximation function $\psi_w(0;\theta,q) = \frac{\theta\Gamma(2/q)}{\Gamma(1/q)} > 0$, which does tend to zero as $\theta \to 0$. In order to deal with this , one omit the second term of (3.7), analogous to the method in [24]. Hence, our WICA is rewritten as

$$\psi_w^\star(x_j;\theta,q) = \psi_w(x_j;\theta,q) - \frac{\theta\Gamma(2/q)}{\Gamma(1/q)}\{1 - \mathrm{erf}_{\frac{q}{2}}(\frac{(w_j x_j)^2}{\theta^2})\} = w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}),$$

and $\psi_w^\star(0;\theta,q) = 0$. On account of $\frac{d}{dx}\mathrm{erf}_q(x) = \frac{q\exp(-x^q)}{\Gamma(1/q)} > 0$ that means $\mathrm{erf}_q(x)$ is a monotone increasing function, then the subtracted term is a monotone decreasing function and only has much effect for $w_j x_j$ close to zero. Thus this treatment is reasonable and feasible.

The following proposition depicts the properties of WHA and WICA.

Proposition 3.7. *For any $x_j \in \mathbb{R}$ and $q \in \mathbb{E}_+$, we have* *(a)* $\lim_{\theta\to 0} h_w(x_j;\theta) = |w_j x_j|$ *and* $\lim_{\theta\to 0}\psi_w^\star(x_j;\theta,q) = |w_j x_j|$*;*

(b) $\lim_{\theta\to\infty} h_w(x_j;\theta) = 0$ *and* $\lim_{\theta\to\infty}\psi_w^\star(x_j;\theta,q) = 0$*;*

(c) $0 \le h_w(x_j;\theta) \le |w_j x_j|$ *and* $0 \le \psi_w^\star(x_j;\theta,q) \le |w_j x_j|$ *for fixed* θ*;*

(d) $\psi_w^\star(x_j;\theta,q)$"$\geq$"$h_w(x_j;\theta)$ *for fixed* θ*, where* "$\geq$" *means to approximate the absolute value function better.*

Proof. It is obvious that (a), (b) and (c) hold for any $x_j \in \mathbb{R}$ and $q \in \mathbb{E}_+$. In order to prove (d) we introduce the distance of two real-valued function f and g as

$$\|f - g\|_\infty = \max_{x\in\mathbb{R}}\{f(x) - g(x)\}.$$

Notice that for fixed θ,

$$\lim_{x_j\to\infty} |h_w(x_j;\theta) - |w_j x_j|| = \frac{\theta}{2} \;\; \textit{and} \;\; h_w(0;\theta) = 0.$$

Moreover, when $x_j \geq 0$ and $q \in \mathbb{E}_+$, combining the fact that $0 < \mathrm{erf}_q(x) < 1$ and the inequality by Alzer [2] as follows

$$\mathrm{erf}_q(x) \geq \{1 - \exp(-x^q)\}^{1/q},$$

we have

$$\begin{aligned}
|\psi_w^\star(x_j;\theta,q) - |w_j x_j|| &= |w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}) - w_j x_j| \\
&= |w_j x_j\{\mathrm{erf}_q(\frac{w_j x_j}{\theta}) - 1\}| \\
&= w_j x_j\{1 - \mathrm{erf}_q(\frac{w_j x_j}{\theta})\} \\
&\leq w_j x_j\{1 - (1 - \exp(-(w_j x_j)^q))^{1/q}\} \\
&\leq w_j x_j \cdot \exp\{-\frac{(w_j x_j)^q}{q}\},
\end{aligned}$$

where the last inequality is established by exploiting that $(a-b)^{1/q} \geq a^{1/q} - b^{1/q}$ for $a \geq b \geq 0$ and $q > 1$. Hence, we obtain

$$\lim_{x_j\to\infty} |\psi_w^\star(x_j;\theta,q) - |w_j x_j|| \leq \lim_{x_j\to\infty} w_j x_j \cdot \exp\{-\frac{(w_j x_j)^q}{q}\} = 0,$$

and then

$$\lim_{x_j\to\infty} |\psi_w^\star(x_j;\theta,q) - |w_j x_j|| = 0 \;\; \textit{and} \;\; \psi_w^\star(0;\theta,q) = 0.$$

So

$$\max_{x_j\in\mathbb{R}} |h_w(x_j;\theta) - |w_j x_j|| = \frac{\theta}{2}, \; \max_{x_j\in\mathbb{R}} |\psi_w^\star(x_j;\theta,q) - |w_j x_j|| = 0.$$

Therefore, we get

$$\|\psi_w^\star(x_j;\theta,q) - |w_j x_j|\|_\infty < \|h_w(x_j;\theta) - |w_j x_j|\|_\infty,$$

which shows that $\psi_w^\star(x_j;\theta,q)$ best approximates the function $|w_j x_j|$. This conclusion is amply illustrated in Fig.3.5. □

We now consider the scalar penalty. In 2017, Selesnick proposed the minimax-concave penalty (MCP) function [18] based on the Huber approximation (HA). Its weighted version is shown below:

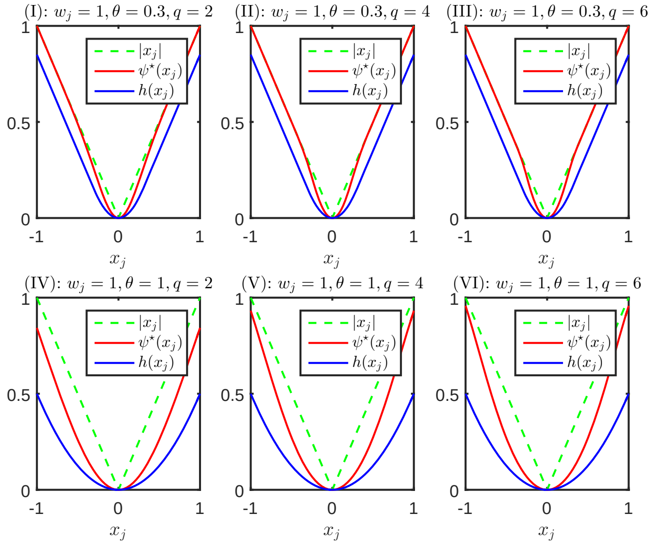


Fig. 3.5: The weighted-based integral convolution approximate (WICA) vs the weighted-based Huber approximation (WHA) in the cases with varying parameters.

Definition 3.8. *The weighted-based minimax-concave penalty (WMCP) function is defined as*

$$\Phi_w(x_j;\theta) := |w_j x_j| - h_w(x_j;\theta) = \begin{cases} |w_j x_j| - \dfrac{(w_j x_j)^2}{2\theta}, & |w_j x_j| \leq \theta; \\ \dfrac{\theta}{2}, & |w_j x_j| \geq \theta. \end{cases}$$

By the same token, we can share the idea and provide an innovative penalty on the strength of the weighted-based convolution approximate (WICA).

Definition 3.9. *The weighted-based integral convolution penalty (WICP) function is defined as*

$$\varphi_w(x_j;\theta,q) := |w_j x_j| - \psi^{\star}_w(x_j;\theta,q) = |w_j x_j| - w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}).$$

Remark 3.10. *In Fig.3.6, we offer the relationship among the absolute value function, WMCP and WICP. It is obvious that of the three penalties the WICP as a function of x_j increases the slowest for the fixed value of θ and q, which means the WICP becomes more concave at $x_j = 0^+$ and induces sparsity more strongly. Meanwhile, we find that the absolute value function excessively penalize the big x_j and the WMCP mildly penalize the large value, but it can't avoid to underestimate high-amplitude components in signal recovery. It's surprising that our method can hardly punish the large component such that the shortcomings of the former two are eliminated.*

3.2 Convexity analysis

In this section, we replace the absolute value function of (3.5) by the WICP as a regularization term and briefly discuss the convexity of the new objective function and present a convexity condition.

Proposition 3.11. *Let $\lambda, w_j > 0$ and $q \in \mathbb{E}_+$. The convexity of objective with weighted-based integral convolution penalty (WICP) function as follows*

$$J_w(x_j;\lambda) = \frac{1}{2}(y_i - a_{ij}x_j)^2 + \lambda\varphi_w(x_j;\theta,q) \tag{3.8}$$

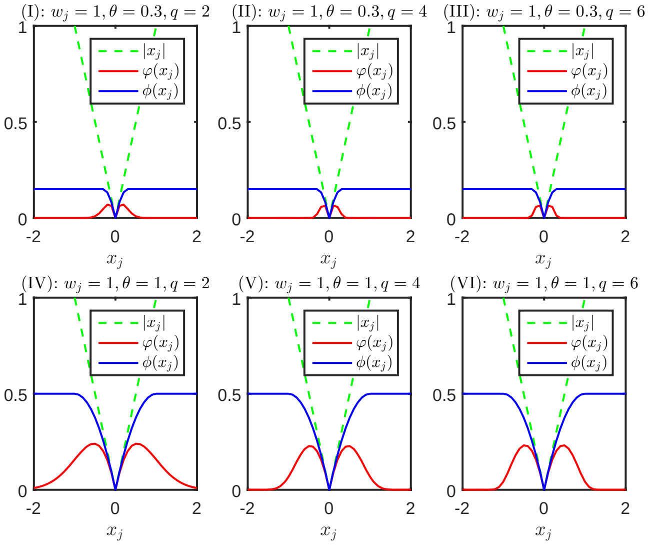


Fig. 3.6: The weighted-based integral convolution penalty (WICP) function vs the weighted-based minimax-concave penalty (WMCP) function in the cases with varying parameters.

is guaranteed provided

$$w_j^2 \leq \frac{\theta\Gamma(1/q)}{2\lambda q} a_{ij}^2.$$

Remark 3.12. *Given a_{ij} and w_j, the convexity condition is easily achieved by tuning the value of the scale parameter θ and the shape parameter q. In other words, the objective function* (3.8) *has more freedom of conversion between convex and nonconvex. If* (3.8) *is convex, the solution will not fall into the local minima.*

Proof.

$$J_w(x_j;\lambda) = \frac{1}{2}(y_i - a_{ij}x_j)^2 + \lambda\varphi_w(x_j;\theta,q) = \frac{1}{2}(y_i - a_{ij}x_j)^2 - \lambda w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta}) + \lambda|w_j x_j|.$$

Obviously, $\lambda|w_j x_j|$ is convex for $\lambda > 0$. Thus it suffices to prove that

$$S = \frac{1}{2}(y_i - a_{ij}x_j)^2 - \lambda w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta})$$

is convex. That is to say, we need to prove that the second derivative of S is nonnegative. Hence, we have

$$\frac{d}{dx_j}S = a_{ij}^2 x_j - y_i a_{ij} - \lambda w_j\{\mathrm{erf}_q(\frac{w_j x_j}{\theta}) + \frac{q w_j x_j}{\theta\Gamma(1/q)}\exp(-(\frac{w_j x_j}{\theta})^q)\},$$

so

$$\begin{aligned}
\frac{d^2}{dx_j^2}S &= a_{ij}^2 - \frac{2\lambda q w_j^2}{\theta\Gamma(1/q)}\{\exp(-(\frac{w_j x_j}{\theta})^q) - \frac{q}{2}(\frac{w_j x_j}{\theta})^q \exp(-(\frac{w_j x_j}{\theta})^q)\} \\
&= a_{ij}^2 - \frac{2\lambda q w_j^2}{\theta\Gamma(1/q)} \cdot \frac{1 - \frac{q}{2}(\frac{w_j x_j}{\theta})^q}{\exp\{(\frac{w_j x_j}{\theta})^q\}} \\
&\geq a_{ij}^2 - \frac{2\lambda q w_j^2}{\theta\Gamma(1/q)} \cdot \frac{1 - \frac{q}{2}(\frac{w_j x_j}{\theta})^q}{1 + (\frac{w_j x_j}{\theta})^q} \\
&\geq a_{ij}^2 - \frac{2\lambda q w_j^2}{\theta\Gamma(1/q)}, \qquad (3.9)
\end{aligned}$$

where the first inequality is founded by $e^x \geq 1+x$ and the second inequality follows from the fact that $\frac{1-\frac{q}{2}x^q}{1+x^q} \leq 1$ for $q \in \mathbb{E}_+$. Therefore, S is convex if $\frac{d^2}{dx_j^2}S \geq a_{ij}^2 - \frac{2\lambda q w_j^2}{\theta\Gamma(1/q)} \geq 0$. □

4 Penalty Function in Vector Case

In this section, We extend the univariate WICP to the multivariate penalty function which is used as a regularization term.

4.1 Multivariate penalty

In term of compressed sensing, the null space property [5] is an important sufficient and necessary condition for the exact recovery of any sparse signal. This condition can be utilized if and only if the corresponding function satisfies the separability, for instance, $\|\boldsymbol{x}\|_1 = \sum_{j=1}^n |x_j|$. Because the generalized weighted-based Huber approximation has no a simple explicit expression and the separability, so obviously its analysis is unnecessary and more difficulty. Here we only present our penalty function according to the previous method.

Definition 4.1. *The generalized weighted-based integral convolution approximate (g-WICA) of ℓ_1-norm is defined as*

$$\psi_w(\boldsymbol{x};\theta,q) := \sum_{j=1}^n \{w_j x_j \cdot \mathrm{erf}_q(\frac{w_j x_j}{\theta})\}.$$

There are some evident properties of the g-WICA as follows:

Proposition 4.2. *For any $\boldsymbol{x} \in \mathbb{R}^n$ and $q \in \mathbb{E}_+$, we have*

(a) $\lim_{\theta \to 0} \psi_w(\boldsymbol{x};\theta,q) = \|W\boldsymbol{x}\|_1$*;*

(b) $\lim_{\theta \to \infty} \psi_w(\boldsymbol{x};\theta,q) = 0$*;*

(c) $0 \leq \psi_w(\boldsymbol{x};\theta,q) \leq \|W\boldsymbol{x}\|_1$ *for fixed θ;*

We put forward a satisfactory penalty function to replace ℓ_1-norm penalty by exploiting the difference between two convex functions based on the g-WICA.

Definition 4.3. *The generalized weighted-based integral convolution penalty (g-WICP) function is defined as*

$$\Phi_w(\boldsymbol{x};\theta,q) := \|W\boldsymbol{x}\|_1 - \psi_w(\boldsymbol{x};\theta,q).$$

4.2 Convexity analysis

Proposition 4.4. *Let $\lambda > 0$ and $q \in \mathbb{E}_+$. The convexity of objective with generalized weighted-based integral convolution penalty (g-WICP) function as follows*

$$J_w(\boldsymbol{x};\lambda) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 + \lambda\Phi_w(\boldsymbol{x};\theta,q)$$

is guaranteed provided

$$W^TW \preccurlyeq \frac{\theta\Gamma(1/q)}{2\lambda q}A^TA. \tag{4.10}$$

Remark 4.5. *Notice that the convexity condition* (4.10) *is equivalent to saying that* $\boldsymbol{x}^T\{A^TA-\frac{2\lambda q}{\theta\Gamma(1/q)}W^TW\}\boldsymbol{x} \geq 0$ *holds for the real symmetric matrices* A^TA*,* W^TW *and any nonzero vector* $\boldsymbol{x}$*. Also because* $\boldsymbol{x}^T\{A^TA - \frac{2\lambda q}{\theta\Gamma(1/q)}W^TW\}\boldsymbol{x} = \|A\boldsymbol{x}\|_2^2 - \frac{2\lambda q}{\theta\Gamma(1/q)}\|W\boldsymbol{x}\|_2^2$*, where* $\|A\boldsymbol{x}\|_2^2 \geq 0$ *with equality holds if and only if* $\boldsymbol{x} \in \ker A$ *and* $\|W\boldsymbol{x}\|_2^2 > 0$*, so it could hardly reach to the convexity condition. But we can control the scale parameter* θ *and the shape parameter* q *such that* $\boldsymbol{x}^T\{A^TA - \frac{2\lambda q}{\theta\Gamma(1/q)}W^TW\}\boldsymbol{x} \geq -\iota$ *(*ι *is rather small). Specially, from the property (b) of proposition 4.2, the g-WICP reduces to* $\|W\boldsymbol{x}\|_1$ *as* $\theta \to \infty$*, then* $J_w(\boldsymbol{x};\lambda)$ *will become a convex function. In summary, the convexity condition could be almost satisfied.*

Proof.

$$J_w(\boldsymbol{x};\lambda) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 + \lambda\Phi_w(\mathbf{x};\theta,q) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 - \lambda\sum_{j=1}^{n}\{w_jx_j \cdot \mathrm{erf}_q(\frac{w_jx_j}{\theta})\} + \lambda\|W\boldsymbol{x}\|_1.$$

Obviously, $\lambda\|W\boldsymbol{x}\|_1$ is convex when $\lambda > 0$. Thus it suffices to prove that

$$S = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 - \lambda\sum_{j=1}^{n}\{w_jx_j \cdot \mathrm{erf}_q(\frac{w_jx_j}{\theta})\}$$

is convex. That is to say, we need to prove that the Hessian matrix of S is nonnegative definite. Hence, we have

$$\nabla_x S = A^T(A\boldsymbol{x} - \boldsymbol{y}) - \lambda\left\{w_j\mathrm{erf}_q(\frac{w_jx_j}{\theta}) + \frac{qw_j^2x_j}{\theta\Gamma(1/q)}\exp(-(\frac{w_jx_j}{\theta})^q)\right\}_{j=1}^{n},$$

so

$$\nabla_x^2 S = A^TA - \frac{2\lambda q}{\theta\Gamma(1/q)}W^TW \cdot \mathrm{Diag}\{\exp(-(\frac{w_jx_j}{\theta})^q) - \frac{q}{2}(\frac{w_jx_j}{\theta})^q\exp(-(\frac{w_jx_j}{\theta})^q)\}.$$

It follows from the proof of (3.9) that

$$\mathrm{Diag}\{\exp(-(\frac{w_jx_j}{\theta})^q) - \frac{q}{2}(\frac{w_jx_j}{\theta})^q\exp(-(\frac{w_jx_j}{\theta})^q)\} \preccurlyeq I_n,$$

then

$$\nabla_x^2 S \succcurlyeq A^TA - \frac{2\lambda q}{\theta\Gamma(1/q)}W^TW.$$

Hence, if $A^TA \succcurlyeq \frac{2\lambda q}{\theta\Gamma(1/q)}W^TW$ that means $\nabla_x^2 S$ is nonnegative definite, then S is convex. □

5 Numerical simulations

In this section, we provide an efficient algorithm and a series of numerical simulations to evaluate the performance of our model (1.2) and (1.4) for signal recovery.

5.1 Optimization Algorithm

Here, we first consider the optimization algorithm for solving the unweighted version problem (1.2) and then extend it to the weighted case. It is worth to notice that we can rewrite the objective function in (1.2) as follows

$$J(\boldsymbol{x};\lambda) = \frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2 + \lambda\Phi(\boldsymbol{x};\theta,q) = \left(\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1\right) - \lambda\sum_{j=1}^{n}\{x_j\cdot \mathrm{erf}_q(\frac{x_j}{\theta})\} = C_1(\boldsymbol{x}) - C_2(\boldsymbol{x}), \quad (5.11)$$

where $C_1(\boldsymbol{x})$ and $C_2(\boldsymbol{x})$ are proper lower semicontinuous convex function. The form (5.11) is known as a DC decomposition of the function $J(\boldsymbol{x};\lambda)$, i.e. difference of convex functions. Corresponsively, there exists a difference of convex functions algorithm (DCA) for solving this kind of problem. The core idea of the DCA is to search for the optimal solution of $J(\boldsymbol{x};\lambda)$ iteratively via the following two steps

$$\begin{cases} \boldsymbol{z}^{(\upsilon)} \in \partial C_2(\boldsymbol{x}^{(\upsilon)}), \\ \boldsymbol{x}^{(\upsilon+1)} = \arg\min\limits_{\boldsymbol{x}\in\mathbb{R}^n} C_1(\boldsymbol{x}) - (C_2(\boldsymbol{x}^{(\upsilon)}) + \langle \boldsymbol{z}^{(\upsilon)}, \boldsymbol{x}-\boldsymbol{x}^{(\upsilon)}\rangle). \end{cases}$$

First, the subgradient sequences $\{\boldsymbol{z}^{(\upsilon)}\}$ of the convex function $C_2(\boldsymbol{x})$ at the $\boldsymbol{x}^{(\upsilon)}$ is calculated and then we obtain the sequences $\{\boldsymbol{x}^{(\upsilon)}\}$ that reduces monotonically the objective function $J(\boldsymbol{x};\lambda)$ by utilizing the definition of the subgradient. From the previous analysis we can see that $C_2(\boldsymbol{x}) = \lambda\sum_{j=1}^{n}\{x_j\cdot\mathrm{erf}_q(\frac{x_j}{\theta})\}$ is differentiable with gradient $\boldsymbol{z} = \lambda\big\{\mathrm{erf}_q(\frac{x_j}{\theta}) + \frac{qx_j}{\theta\Gamma(1/q)}\exp(-(\frac{x_j}{\theta})^q)\big\}_{j=1}^{n}$, thus we have

$$\boldsymbol{x}^{(\upsilon+1)} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n}\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2 + \langle\boldsymbol{\alpha}^{(\upsilon)},\boldsymbol{x}\rangle + \lambda\|\boldsymbol{x}\|_1 + \rho,$$

where $\rho = -\lambda\sum_{j=1}^{n}\frac{q({x_j}^{(\upsilon)})^2}{\theta\Gamma(1/q)}\exp(-(\frac{{x_j}^{(\upsilon)}}{\theta})^q)$ and $\boldsymbol{\alpha}^{(\upsilon)} = -\boldsymbol{z}^{(\upsilon)}$ can be treated as a constant and a constant vector at $(\upsilon+1)$-th iteration, respectively. Therefore, we turn to solve the following modified optimization problem

$$\boldsymbol{x}^{(\upsilon+1)} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n}\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2 + \langle\boldsymbol{\alpha}^{(\upsilon)},\boldsymbol{x}\rangle + \lambda\|\boldsymbol{x}\|_1. \quad (5.12)$$

The form (5.12) is used as outer iteration with a termination criterion $\frac{\|\boldsymbol{x}^{(\upsilon+1)}-\boldsymbol{x}^{(\upsilon)}\|_2}{\max(\|\boldsymbol{x}^{(\upsilon)}\|_2,1)} \le \epsilon$ for a preset moderately small threshold ϵ. Because of its separable structure, problem (5.12) can be efficiently solved by the alternating direction method with multipliers (ADMM), which decomposes the original joint minimization problem into two easily solved subproblems. More specifically, we introduce an auxiliary variable $\boldsymbol{\beta}\in\mathbb{R}^n$, the problem (5.12) can be equivalently reformulated as

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\underbrace{\frac{1}{2}\|\boldsymbol{y}-A\boldsymbol{x}\|_2^2 + \langle\boldsymbol{\alpha}^{(\upsilon)},\boldsymbol{x}\rangle}_{f_1(\boldsymbol{x})} + \underbrace{\lambda\|\boldsymbol{\beta}\|_1}_{f_2(\boldsymbol{\beta})} \quad s.t. \quad \boldsymbol{x}-\boldsymbol{\beta} = \boldsymbol{0}.$$

The corresponding augmented Lagrangian function is given by

$$\mathcal{L}(\boldsymbol{x},\boldsymbol{\beta},\boldsymbol{\gamma}) = f_1(\boldsymbol{x}) + f_2(\boldsymbol{\beta}) + \boldsymbol{\gamma}^T(\boldsymbol{x}-\boldsymbol{\beta}) + \frac{\tau}{2}\|\boldsymbol{x}-\boldsymbol{\beta}\|_2^2,$$

where τ is a positive scalar for controlling the speed of the algorithm and $\boldsymbol{\gamma}$ is the Lagrangian multiplier vector. It is obviously difficult to solve the above function directly. We can first minimize a variable for fixed the other variables as follows

$$\begin{cases} \mathbf{x}^{(l+1)} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n}\{f_1(\boldsymbol{x}) + \frac{\tau}{2}\|\boldsymbol{x} - \boldsymbol{\beta}^{(l)} + \boldsymbol{\gamma}^{(l)}\|_2^2\}, & (5.13) \\ \boldsymbol{\beta}^{(l+1)} = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^n}\{f_2(\boldsymbol{\beta}) + \frac{\tau}{2}\|\boldsymbol{x}^{(l+1)} - \boldsymbol{\beta} + \boldsymbol{\gamma}^{(l)}\|_2^2\}, & (5.14) \\ \boldsymbol{\gamma}^{(l+1)} = \boldsymbol{\gamma}^{(l)} + \tau(\boldsymbol{x}^{(l+1)} - \boldsymbol{\beta}^{(l+1)}). \end{cases}$$

Computing $\boldsymbol{x}^{(l+1)}$:

Since the corresponding objective function of sub-problem (5.13) is quadratic, and then letting its first-order derivative equal to zero directly yields

$$\boldsymbol{x}^{(l+1)} = (A^T A + \tau I)^{-1}(A^T \boldsymbol{y} - \boldsymbol{\alpha}^{(v)} + \tau \boldsymbol{x}^{(l)} - \boldsymbol{\beta}^{(l)}).$$

Computing $\boldsymbol{\beta}^{(l+1)}$:

As suggested by Hale et al. [11], the solution to the sub-problem (5.14) can be gained as

$$\mathcal{S}_{\frac{\lambda}{\tau}}(\boldsymbol{z}) = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{\beta} - \boldsymbol{z}\|_2^2 + \frac{\lambda}{\tau}\|\boldsymbol{\beta}\|_1,$$

where $\mathcal{S}_{\frac{\lambda}{\tau}}(z_j) = \operatorname{sign}(z_j)\max(|z_j| - \frac{\lambda}{\tau}, 0)$ is the soft-thresholding operator.

The above alternative updating steps as inner iteration are repeated until the convergence condition is satisfied or the number of iteration exceeds a preset threshold. Following the suggestions in [30], when $\|\boldsymbol{x}^{(l)} - \boldsymbol{\beta}^{(l)}\|_2 \leq \sqrt{n}\epsilon^{abs} + \epsilon^{rel}\max(\|\boldsymbol{x}^{(l)}\|_2, \|\boldsymbol{\beta}^{(l)}\|_2)$ and $\|\tau(\boldsymbol{\beta}^{(l)} - \boldsymbol{\beta}^{(l-1)})\|_2 \leq \sqrt{n}\epsilon^{abs} + \epsilon^{rel}\|\boldsymbol{\gamma}^{(l)}\|_2$ with an absolute tolerance ϵ^{abs} and a relative tolerance ϵ^{rel}, an approximate solution of problem (5.12) can be fully guaranteed. All solving processes can be summed up as Algorithm 5.1.

Now we consider the algorithm for solving the weighted version problem (1.4). We add a weighted step on the basis of the solution obtained by Algorithm 5.1, which is similar to the idea of iterative reweighted ℓ_1-minimization algorithm (IRL1) [4]. The proposed simple iterative algorithm that alternates between estimating $\boldsymbol{x}$ and redefining the weights as described in Algorithm 5.2.

In Algorithm 5.2, for the sake of providing at ability and ensuring that a zero-valued component in $\bar{\boldsymbol{x}}^{(v)}$ does not rigidly forbid a nonzero estimate at the next step, the nonnegative parameter ϱ is introduced. Moreover, the w_j-update and ϱ-update are in accordance with suggestions of [13] and $\mathrm{r}(\cdot)_{k+1}$ denotes $(k+1)$-th value of the rearrangement of $|\cdot|$ where k is the sparsity of vector $\boldsymbol{x}$.

5.2 Experimental Results on Noiseless Data

In this section we provide simulations to compare our methods with four state-of-the-art algorithms for sparse recovery: lasso-ADMM [2], which solves the lasso problem (1.1) by ADMM; ℓ_{1-2}-DCA [30] that addresses the unconstrained ℓ_1-ℓ_2 minimization problem based on the DCA; Half thresholding [28] for $\ell_{1/2}$ regularization; and the iteratively reweighted ℓ_1 minimization algorithm (IRL1). All the above-mentioned solvers are coded on a PC with 4 GB of RAM and Intel core i5-4200M (2.5GHz). In order to avoid the randomness, we perform 100 times against each test and report the average result.

Algorithm 5.1 An algorithm for the minimization problem with g-ICP (gICP-DCA)

```
**Input:** $\boldsymbol{y} \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n} (m \ll n)$, $\lambda$, $q$, $\theta$ and $\tau$.
**Output:** $\boldsymbol{x}^\sharp \in \mathbb{R}^n$.
Initialize $\boldsymbol{x}^{(0)} = \boldsymbol{0}$ and $\upsilon = 0$.
// Outer loop
**while** no convergence **do**
  Initialize $\boldsymbol{\beta}^{(0)} = \boldsymbol{0}$, $\boldsymbol{\gamma}^{(0)} = \boldsymbol{0}$ and $l = 0$.
  // Inner loop (ADMM)
  **while** no convergence **do**
    $\boldsymbol{x}^{(l+1)} = (A^T A + \tau I)^{-1}(A^T \boldsymbol{y} - \boldsymbol{\alpha}^{(\upsilon)} + \tau \boldsymbol{x}^{(l)} - \boldsymbol{\beta}^{(l)})$.
    $\boldsymbol{\beta}^{(l+1)} = \mathcal{S}_{\frac{\lambda}{\tau}}(\boldsymbol{x}^{(l+1)} + \boldsymbol{\gamma}^{(l)}/\tau)$.
    $\boldsymbol{\gamma}^{(l+1)} = \boldsymbol{\gamma}^{(l)} + \tau(\boldsymbol{x}^{(l+1)} - \boldsymbol{\beta}^{(l+1)})$.
    $l = l + 1$.
  **end while**
  $\boldsymbol{x}^{(\upsilon+1)} = \boldsymbol{x}^{(l)}$.
  Update $\boldsymbol{\alpha}^{(\upsilon)}$.
  $\upsilon = \upsilon + 1$.
**end while**
**return** $\boldsymbol{x}^\sharp = \boldsymbol{x}^{(\upsilon)}$.
```

5.2.1 Experimental Setup

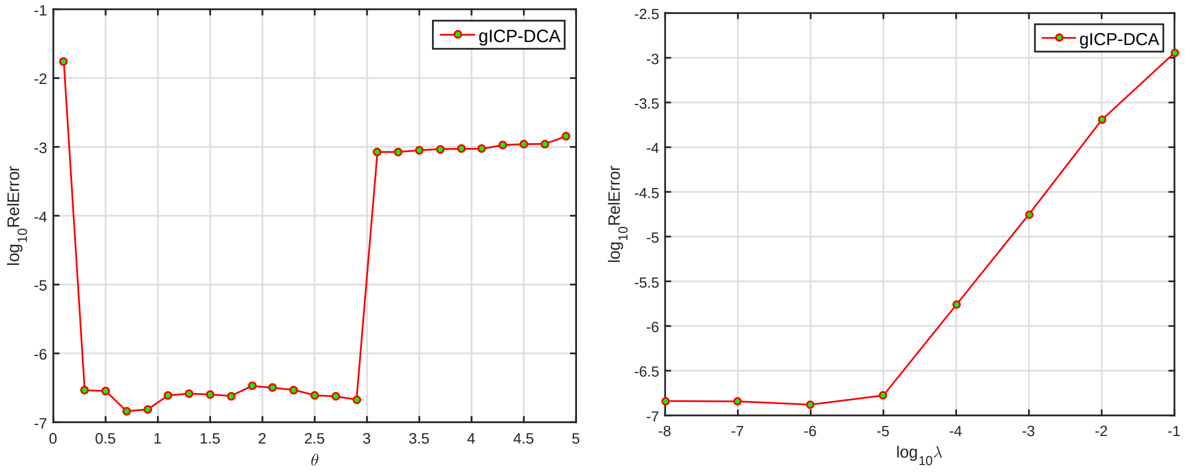
In the absence of noise, we generate a measurement matrix $A \in \mathbb{R}^{64 \times 256}$ with entries drawn independently from the standard Gaussian distribution, Such measurement matrix A is known to satisfy (with high probability) the RIP . The locations of nonzero components of the signals $\boldsymbol{x} \in \mathbb{R}^{256}$ are uniformly randomly generated, and the nonzero values are chosen from a Gaussian distribution with mean 0 and standard deviation 1. Given A and $\boldsymbol{x}$, the measurements $\boldsymbol{y}$ are produced by $\boldsymbol{y} = A\boldsymbol{x}$. We deem that the signal $\boldsymbol{x}^\sharp$ can be as a successful reconstruction for the original signal $\boldsymbol{x}$ from the measurements $\boldsymbol{y}$ if the relative error (abbreviation: RelError) meets $\|\boldsymbol{x}^\sharp - \boldsymbol{x}\|_2 / \|\boldsymbol{x}\|_2 < 10^{-3}$.

In our experiments, the parameters of lasso-ADMM, Half thresholding, IRL1 and ℓ_{1-2}-DCA are tuned to achieve the best performance. For gICP-DCA and gWICP-DCA, we set tolerances both $\epsilon = 10^{-2}$, $\epsilon^{abs} = 10^{-7}$, and $\epsilon^{rel} = 10^{-5}$. Specially, we set the number of outer and inter loop in Algorithm 5.1 to be 10 and 5000, respectively. For gWICP-DCA, we set the maximum number of iterations to be 20 and see that much of the benefit comes from few reweighting iterations, and so the added computational cost for improved signal recovery is quite moderate. In addition, for gICP-DCA and gWICP-DCA, the shape parameter q of GGF is empirically set to be 2 (maybe it's because the elements of the signal $\boldsymbol{x}$ are subject to the Gauss distribution). Fixed q, we employ a "trial and error" strategy to choose the scale parameter θ and the regularization parameter λ (set the penalty parameter $\tau = 10\lambda$). As illustrated in Fig.5.7, gICP-DCA reaches the minimum relative error when $\theta = 0.7$, $\lambda = 10^{-8}$ and $\tau = 10^{-7}$ (in the noiseless case where λ should be always set to a small number such that

Algorithm 5.2 An algorithm for the minimization problem with g-WICP (gWICP-DCA)

```
Input: $\boldsymbol{y} \in \mathbb{R}^m$, $A \in \mathbb{R}^{m\times n} (m \ll n)$, $\lambda$, $q$, $\theta$ and $\tau$.
Output: $\boldsymbol{x}^\sharp \in \mathbb{R}^n$.
Initialize $A^{(0)} = A$, $W^{(0)} = I$, $\boldsymbol{x}^{(0)} = \boldsymbol{0}$ and $\upsilon = 0$.
while no convergence do
    $\tilde{A}^{(\upsilon+1)} = A^{(\upsilon)}(W^{(\upsilon)})^{-1}$.
    $\bar{\boldsymbol{x}}^{(\upsilon+1)} = \text{gICP-DCA}(\tilde{A}^{(\upsilon)}, \boldsymbol{y}, \lambda, q, \theta, \tau)$.
    $\boldsymbol{x}^{(\upsilon+1)} = (W^{(\upsilon)})^{-1}\bar{\boldsymbol{x}}^{(\upsilon+1)}$.
    Update weights $w_j^{(\upsilon+1)} = \{(x_j{}^{(\upsilon)})^2 + (\varrho^{(\upsilon)})^2\}^{-\frac{1}{2}}$.
    $\varrho^{(\upsilon)} = \min\{\varrho^{(\upsilon)}, c \cdot \mathrm{r}(\boldsymbol{x}^{(\upsilon+1)})_{k+1}\}$ where $c \in (0,1)$ is a constant.
    $\upsilon = \upsilon + 1$.
end while
return $\boldsymbol{x}^\sharp = \bar{\boldsymbol{x}}^{(\upsilon)}$.
```

ℓ_2-norm data fidelity possesses a slight effect). Here gWICP-DCA shares same parameters with gICP-DCA.



(a)

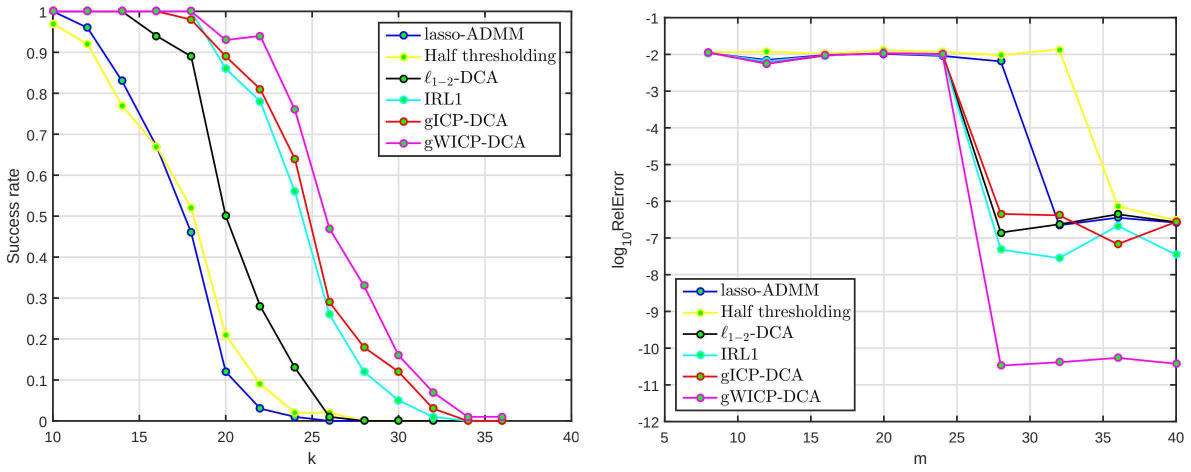
(b)

Fig. 5.7: Parameters selection for gICP-DCA with sparsity $k = 20$. (a) for θ versus RelError with $\lambda = 10^{-8}$ and $\tau = 10^{-7}$. (b) for λ versus RelError with $\theta = 0.7$.

5.2.2 Results for RIP Matrix

In this section, we implement two sets of experiments for the Gaussian measurement matrix satisfied RIP. Fig.5.8 (a) depicts the relationship between sparsity k and success rate (the success ratio of 100 experiments). It is easy to see that an increasing k leads to a poor reconstruction, however gWICP-DCA owns the highest success rate at various sparsity levels and followed by gICP-DCA. The performance of IRL1 is slightly lower than that of gICP-DCA. The gap between remaining three algorithms and our solvers is pretty big.

Fig.5.8 (b) shows the relative error performance versus the number of measurements m for a signal with length $n = 256$ and sparsity $k = 5$. We notice that when $m \leq 25$, there are no obvious differences. Comparing the gap between these algorithms for $m > 25$, we remark that gWICP-DCA has the lowest relative error and is about three orders of magnitude different from other methods. In a word, the sampling number required for gWICP-DCA is the least in the same case.



(a) (b)

Fig. 5.8: Comparison of algorithms using RIP matrix. (a) for k versus Success rates with sampling number $m = 64$. (b) for m versus RelError with sparsity $k = 5$.

5.2.3 Results for Highly Coherent Matrix

In addition, we also test a more ill-posed and significantly higher coherence sensing matrix which is generated by creating a randomly oversampled partial DCT matrix $A \in \mathbb{R}^{m\times n}$ with $A_i = \cos(2(i-1)\pi\varpi/F)/\sqrt{m}$, where the elements in $\varpi \in \mathbb{R}^n$ draw independently from the uniform distribution and F is a positive integer that is closely related to the coherence of A. In general, a bigger F corresponds to a larger coherence. The test signal is produced by creating an k-sparse signal $\boldsymbol{x}$ whose k entries are independently sampled from a standard normal distribution and located at supports drawn uniformly in $\{1, 2, \cdots, n\}$. Similar to [30], we restricted the elements i, j in supp$(\boldsymbol{x})$ to satisfy $\min_{i\neq j} |i - j| \geq 2F$.

Fig.5.9 provides an example with $A \in \mathbb{R}^{50\times1000}$ and $F = 14$. It's remarkable that such an ill-posed and inverse problem is an enormous challenge for the vast majority of the algorithms, however, our gWICP-DCA and ℓ_{1-2}-DCA still maintain a high recovery success rate for different sparsity levels with gICP-DCA in hot pursuit. In contrast, other algorithms encounter a general performance, especially Half thresholding is almost incapable of signal recovery in the case of a highly coherent measurement matrix.

5.3 Experimental Results on Noise Data

In this section, we conduct some numerical experiments to support the robustness of our solvers in the presence of noise. We compare the performance of our two methods with the other representative algorithms including lasso-ADMM and ℓ_{1-2}-DCA and Half thresholding. Here we do not take IRL1 into consideration

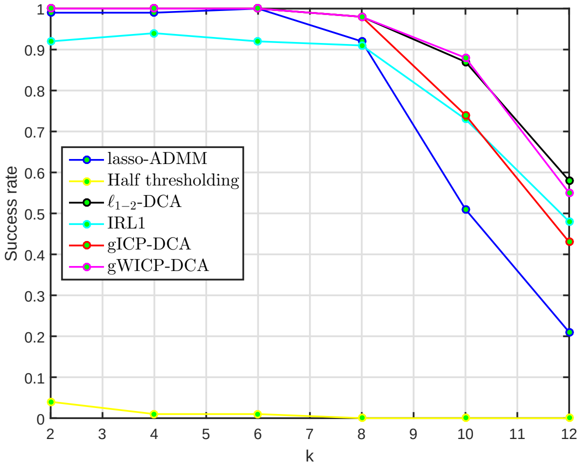
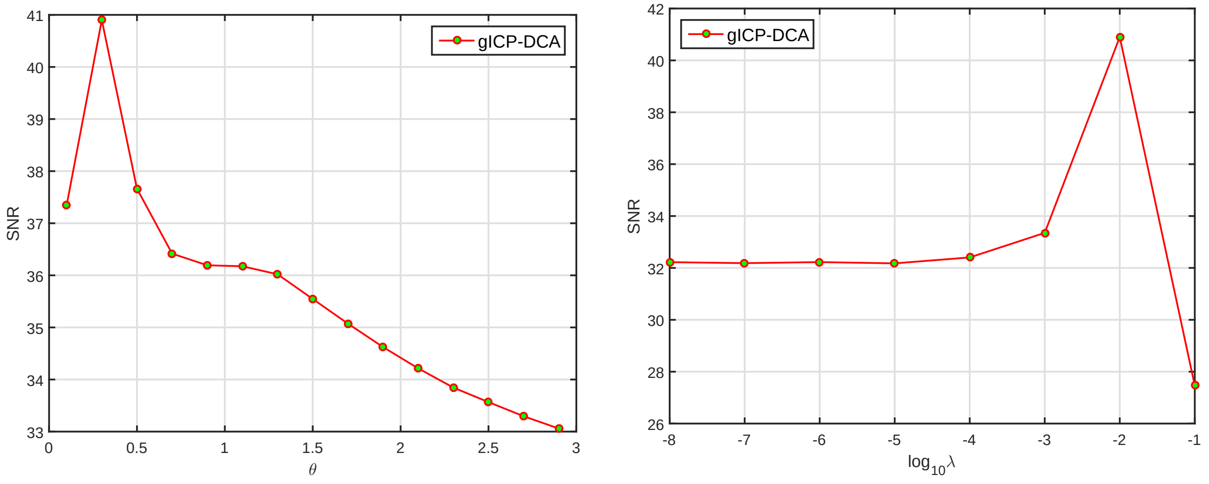


Fig. 5.9: Comparison of algorithms using highly coherent matrix for k versus Success rates with $m = 50$, $n = 1000$ and $F = 14$.

in the comparison, because IRL1 is often used to solve constrained noiseless signal recovery problems, so it is not appropriate in this example. Throughout the rest of experiments, all settings are consistent with the corresponding experiments in 5.2 Section except the scale parameter θ, the regularization parameter λ and the penalty parameter τ (to be chosen later). Using $\boldsymbol{x}$ and A, the measurements $\boldsymbol{y}$ are produced by $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$, where $\boldsymbol{e}$ is the Gaussian white noise with mean 0 and standard deviation 0.05. We uniformly evaluate the recovery performance of all the methods by signal-to-noise ratio (SNR) defined as $20\log(\|\boldsymbol{x}\|_2/\|\boldsymbol{x} - \boldsymbol{x}^\sharp\|_2)$ in decibels (dB).

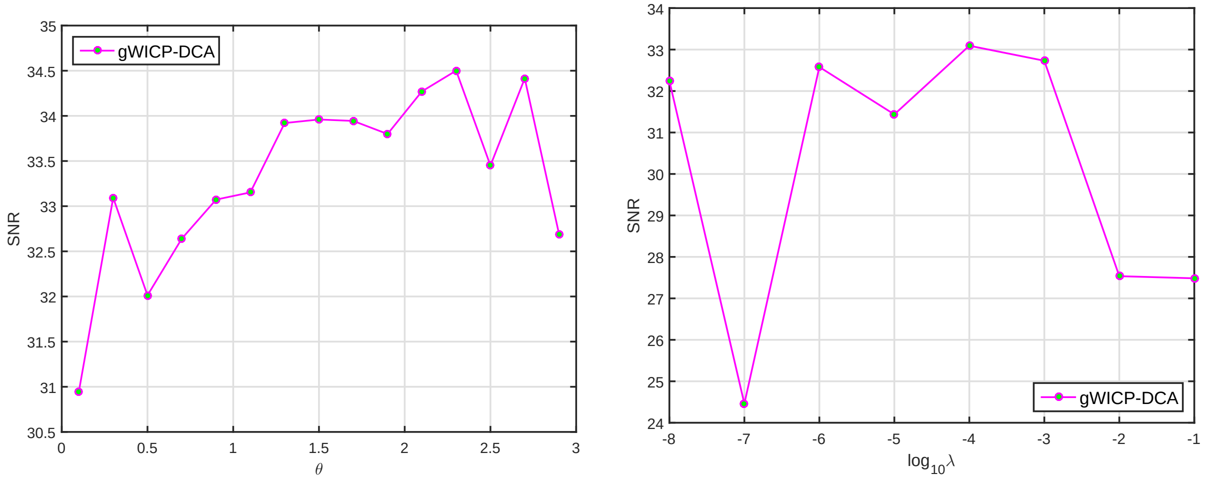
In order to find the better θ, λ, τ that obtain the maximal SNR, two sets of trails have been conducted based on the "trial and error" strategy. From Fig.5.10 and Fig.5.11, the parameters $\theta = 0.3, \lambda = 10^{-2}, \tau = 10^{-1}$ for gICP-DCA and $\theta = 2.4, \lambda = 10^{-4}, \tau = 10^{-3}$ for gWICP-DCA are two good choices.



(a)

(b)

Fig. 5.10: Parameters selection for gICP-DCA with sparsity $k = 10$. (a) for θ versus SNR with $\lambda = 10^{-2}$ and $\tau = 10\lambda$. (b) for λ versus SNR with $\theta = 0.3$.



(a)

(b)

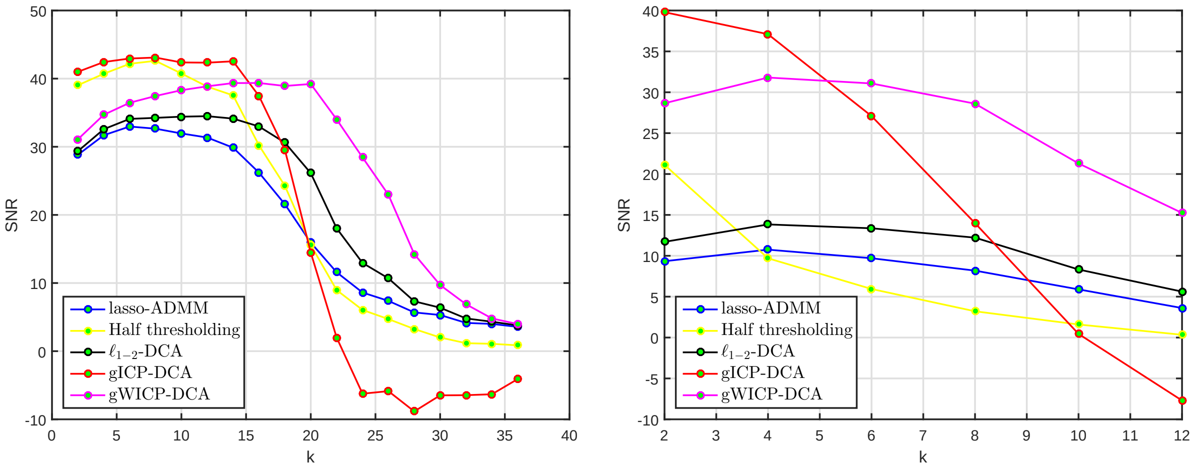
Fig. 5.11: Parameters selection for gWICP-DCA with sparsity $k = 10$. (a) for θ versus SNR with $\lambda = 10^{-4}$ and $\tau = 10\lambda$. (b) for λ versus SNR with $\theta = 2.4$.

5.3.1 Results for RIP Matrix and Highly Coherent Matrix

In this section, under the condition that measurement matrix A satisfies RIP and high coherence, respectively, we study the SNR performance of methods influenced by sparsity levels k. According to Fig.5.12 (a), when $k \leq 15$, all the methods tend to perform better. However, our gICP-DCA and Half thresholding methods rand first and second, respectively, followed by the gWICP-DCA, ℓ_{1-2}-DCA and lasso-ADMM in order. Besides, when sparsity k increases and $k > 15$ or in order words, the difficulty of signal recovery increases, our gWICP-DCA is obviously superior to the other solvers. One can easily see from Fig.5.12 (b) that our proposed two methods behave better than the rest of the methods on the whole. In particular, our gWICP-DCA shows stability for different k's and the gap between other algorithms and our methods is quite clear.

6 Conclusion

This work introduced a robust formulation for signal recovery, which employs two key ideas that 1) the integral convolution approximation of the absolute value function; 2) the difference between two convex functions. Moreover, we generalized the result of the univariate to the multivariate case. Two efficient algorithms, i.e., gICP-DCA and gWICP-DCA, have been provided to solve this criterion. To verify the effectiveness and robustness of our methods, several sets of contrast tests include noise and noiseless case as well as RIP matrix and highly coherent matrix case have been implemented. The obtained results show that our methods outperform other state-of-the-art methods in recovering signals. Some extensions of our work are of interest. For example, this formulation can be extended to structured sparse signal recovery and non-Gaussian noise modeling where the data fidelity term is measured by ℓ_p-norm ($p \neq 2$).



(a)

(b)

Fig. 5.12: (a) Comparison of algorithms using RIP matrix for sparsity k versus SNR with $m = 64, n = 256$. (b) Comparison of algorithms using highly coherent matrix for sparsity k versus SNR with $m = 50, n = 1000$.

Acknowledgment

This work was supported by Natural Science Foundation of China (Grant Nos. 61673015, 61273020), Fundamental Research Funds for the Central Universities (Grant Nos. XDJK2015A007), Science Computing and Intelligent Information Processing of GuangXi higher education key laboratory (Grant Nos. GXSCIIP201702).

References

[1] Heinz H Bauschke, Regina Burachik, Patrick L Combettes, Veit Elser, D Russell Luke, and Henry Wolkowicz. *Fixed-point algorithms for inverse problems in science and engineering*, volume 49. Springer Science &amp; Business Media, 2011.

[2] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[3] Emmanuel J Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, 2008.

[4] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14(5):877–905, 2008.

[5] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Compressed sensing and best k-term approximation. *Journal of the American mathematical society*, 22(1):211–231, 2009.

[6] Zdzislaw Denkowski, Stanislaw Migórski, and Nikolaos S Papageorgiou. *An introduction to nonlinear analysis: theory*. Springer Science & Business Media, 2013.

[7] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[8] Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

[9] Cuixia Gao, Naiyan Wang, Qi Yu, and Zhihua Zhang. A feasible nonconvex relaxation approach to feature selection. In *Aaai*, pages 356–361, 2011.

[10] Rafal Goebel. Self-dual smoothing of convex and saddle functions. *Journal of Convex Analysis*, 15(1):179, 2008.

[11] Elaine T Hale, Wotao Yin, and Yin Zhang. Fixed-point continuation for ℓ_1-minimization: Methodology and convergence. *SIAM Journal on Optimization*, 19(3):1107–1130, 2008.

[12] Peter J Huber et al. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[13] Ming-Jun Lai, Yangyang Xu, and Wotao Yin. Improved iteratively reweighted least squares for unconstrained smoothed ℓ_p minimization. *SIAM Journal on Numerical Analysis*, 51(2):927–957, 2013.

[14] Haiyang Li, Qian Zhang, Jigen Peng, et al. Minimization of fraction function penalty in compressed sensing. *arXiv preprint arXiv:1705.06048*, 2017.

[15] Xiaolei Lv, Guoan Bi, and Chunru Wan. The group lasso for stable recovery of block-sparse signal representations. *IEEE Transactions on Signal Processing*, 59(4):1371–1382, 2011.

[16] Saralees Nadarajah. A generalized normal distribution. *Journal of Applied Statistics*, 32(7):685–694, 2005.

[17] Alberto Seeger. Smoothing a nondifferentiable convex function: the technique of the rolling ball. *Rev. Mat. Apl*, 18(1):45–60, 1997.

[18] Ivan Selesnick. Sparse regularization via convex analysis. *IEEE Transactions on Signal Processing*, 65(17):4481–4494, 2017.

[19] Steven W Smith. *The scientist and engineer's guide to digital signal processing.* California Technical Publishing, 1997.

[20] Pham Dinh Tao and Le Thi Hoai An. Convex analysis approach to dc programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997.

[21] Pham Dinh Tao and Le Thi Hoai An. A dc optimization algorithm for solving the trust-region subproblem. *SIAM Journal on Optimization*, 8(2):476–505, 1998.

[22] Marc Teboulle. Entropic proximal mappings with applications to nonlinear programming. *Mathematics of Operations Research*, 17(3):670–690, 1992.

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

[24] Sergey Voronin, Gorkem Ozkaya, and Davis Yoshida. Convolution based smooth approximations to the absolute value function with application to non-smooth regularization. *arXiv preprint arXiv:1408.6795*, 2014.

[25] Jianjun Wang, Jing Zhang, Wendong Wang, and Chanyun Yang. A perturbation analysis of nonconvex block-sparse compressed sensing. *Communications in Nonlinear Science and Numerical Simulation*, 29(1):416–426, 2015.

[26] Wendong Wang, Jianjun Wang, and Zili Zhang. Robust signal recovery with highly coherent measurement matrices. *IEEE Signal Processing Letters*, 24(3):304–308, 2017.

[27] Jinming Wen, Dongfang Li, and Fumin Zhu. Stable recovery of sparse signals via ℓ_p-minimization. *Applied and Computational Harmonic Analysis*, 38(1):161–176, 2015.

[28] Zongben Xu, Xiangyu Chang, Fengmin Xu, and Hai Zhang. $l_{1/2}$ regularization: A thresholding representation theory and a fast solver. *IEEE Transactions on Neural Networks*, 23(7):1013–1027, 2012.

[29] Li Yin and Feng Qi. Some functional inequalities for generalized error function. *Journal of Computational Analysis & Applications*, 25(7), 2018.

[30] Penghang Yin, Yifei Lou, Qi He, and Jack Xin. Minimization of ℓ_{1-2} for compressed sensing. *SIAM Journal on Scientific Computing*, 37(1):A536–A563, 2015.