Newton Greedy Pursuit: a Quadratic Approximation Method for Sparsity-Constrained Optimization

Xiao-Tong Yuan Qingshan Liu
S-mart Lab, College of Information and Control
Nanjing University of Information Science and Technology
{xtyuan, qsliu}@nuist.edu.cn

Abstract

First-order greedy selection algorithms have been widely applied to sparsity-constrained optimization. The main theme of this type of methods is to evaluate the function gradient in the previous iteration to update the non-zero entries and their values in the next iteration. In contrast, relatively less effort has been made to study the second-order greedy selection method additionally utilizing the Hessian information. Inspired by the classic constrained Newton method, we propose in this paper the NewTon Greedy Pursuit (NTGP) method to approximately minimizes a twice differentiable function over sparsity constraint. At each iteration, NTGP constructs a second-order Taylor expansion to approximate the cost function, and estimates the next iterate as the solution of the constructed quadratic model over sparsity constraint. Parameter estimation error and convergence property of NTGP are analyzed. The superiority of NTGP to several representative first-order greedy selection methods is demonstrated in synthetic and real sparse logistic regression tasks.

1. Introduction

In the past two decades, sparsity models have received broad research interests in high-dimensional statistical learning and signal processing with many significant results obtained in theory, algorithm and applications. A fundamental prior assumption of sparsity models is that the datasets need to be processed exhibit certain low-dimensional structure, which can usually been captured by imposing sparsity constraint on the model parameter space. Therefore it is crucial to develop robust and efficient computational procedures to solve the following sparsity-constrained optimization problem:

$$\min_{x \in \mathbb{R}^p} f(x), \quad \text{s.t. } \|x\|_0 \leq k, \tag{1}$$

where $f : \mathbb{R}^p \mapsto \mathbb{R}$ is a smooth convex cost function.

Unfortunately, due to the cardinality constraint, the problem (1) is generally NP-hard even for the quadratic cost function [11]. Thus, one must seek approximate solvers instead. Particularly, the special case of (1) in linear regression models has gained significant attention in Compressed Sensing (CS) [7] area. A large body of greedy algorithms for CS have been proposed including Orthogonal Matching Pursuit (OMP) [15], Compressive Sampling Matching Pursuit (CoSaMP) [12], Iterative Hard Thresholding (IHT) [3], and subspace pursuit [6] to name a few. The main theme of these iterative algorithms is to use the residual error from the previous iteration to successively approximate the positions of non-zero entries and estimate their values.

While the measure of discrepancy, least square error, used in CS models is often desirable for signal processing applications, it is not the appropriate choice for a variety of other applications. For example, in statistical machine learning the log likelihood function is commonly used in logistic regression problems (see [1] and the references therein) and graphical models learning [9]. Thus, it is desirable to develop theory and algorithms that apply to a broader class of sparsity-constrained learning problems as given in (1). To this end forward greedy selection algorithms have been proposed to select out the non-zero entries in a sequential way [14, 16]. To make the greedy selection procedure more adaptive, [18] proposed a forward-backward algorithm which takes backward steps adaptively whenever beneficial. The forward-backward-type method has also been investigated in [9] for sparse graphical models learning tasks. More recently, [1] proposed the Gradient Support Pursuit Method (GraSP) which selects in the current iteration multiple entries and update their values based on the gradient vector in the previous iteration.

The algorithms just mentioned all belong to the category of first-order methods which only evaluate the function value and its gradient in the previous iteration to update the non-zero entries and their values in the next iteration. Although this leads to efficient iterations, using merely first-order information means that these methods typically require a substantial number of iterations to reach an accurate solution. In classic convex optimization, the constrained Newton method (also known as scaled gradient projection) [2] has been shown to converge superlinearly. At each iteration this method computes a descent direction by minimizing, over the original constraints, a second-order Taylor expansion approximation to the objective function. Schmidt *et al.* [13] proposed the Projected Quasi-Newton (PQN) method in which the quadratic approximation is constructed using a L-BFGS [4] update. It has been empirically shown to frequently beat first-order methods in large-scale problems where evaluation of the function is substantially more expensive than projection onto the ℓ_1-norm ball. The computational efficiency of quadratic approximation method has also been demonstrated by [8] in ℓ_1-regularized sparse precision matrix estimation.

Our contribution. Inspired by the efficiency of constrained Newton-type methods, we propose in this paper the NewTon Greedy Pursuit (NTGP) method to approximately solve (1) with twice continuously differentiable function. Our iterative method is based on a two-level strategy. At the outer level, we construct a sequence of ℓ_0-constrained second-order Taylor expansions of the problem; at the inner level, an iterative hard-thresholding algorithm is used to approximately minimize this ℓ_0-constrained quadratic model. We have analyzed the sparse recovery performance of NTGP under proper assumptions; and evaluated its practical performance in sparse logistic regression tasks. Theoretical results and empirical evidence both show that NTGP can lead to substantial gains in accuracy and computational efficiency. To the best of our knowledge, NTGP is the first Newton-type greedy selection method with performance guarantees.

Notation. Given a vector $x \in \mathbb{R}^p$, we denote $[x]_i$ as its ith entry; x_F as the restriction of x to index set F, i.e., $[x_F]_i = [x]_i$ if $i \in F$, and $[x_F]_i = 0$ otherwise; and x_k as the restriction of x to the top k (in magnitude) entries. Let $\|x\|_2 = \sqrt{x^\top x}$ denote the Euclidean norm; $\|x\|_1 = \sum_{i=1}^p |x_i|$ the ℓ_1-norm; and $\|x\|_0$ the number of nonzero of x. Given a matrix $A \in \mathbb{R}^{p \times p}$, let us denote $[A]_{ij}$ its element on the ith row and jth column; $\tilde{A}_F$ the principal submatrix of A with rows and columns indexed in set F; and A_F the restriction of A to index set F, i.e., $[A_F]_{ij} = [A]_{ij}$ if $i, j \in F$, and $[A_F]_{ij} = 0$ otherwise. Provided that $\tilde{A}_F$ is invertible, we define $A_F^{-1} \in \mathbb{R}^{p \times p}$ by $[A_F^{-1}]_F = \tilde{A}_F^{-1}$ and $[A_F^{-1}]_{ij} = 0$ if $i \notin F$ or $j \notin F$. Let $|A|_\infty = \max_{i,j} |[A_{ij}]|$ denote the element-wise ℓ_∞-norm; $|A|_1 = \sum_{i,j} |[A_{ij}]|$ the element-wise ℓ_1-norm; $\|A\|_1 = \max_j \sum_i |[A_{ij}]|$ the matrix ℓ_1-norm; $\|A\|_{Frob} = \sqrt{\sum_{i,j}[A_{ij}]^2}$ the Frobenius norm; $\|A\|_2 = \sup_{\|x\|_2 \le 1} \|Ax\|_2$ the spectral norm; and vect(A) the vectorization of A. Finally, I represents an identity matrix of compatible size.

2. Newton Greedy Pursuit

We start by briefly reviewing the constrained Newton method for convex optimization. Then we introduce NTGP method as an adaption of constrained Newton method to the non-convex problem (1). We will also analyze the sparse recovery and computational performance of NTGP.

2.1. Constrained Newton method

The constrained Newton method [2, §2.3] iteratively minimizes a convex objective f over a convex set Ω, i.e.,

$$\min_x f(x), \qquad \text{s.t. } x \in \Omega. \tag{2}$$

Here function f is assumed to be twice continuously differentiable over Ω. At iteration t, this method approximates the objective around the current iterate $x^{(t)}$ by the second-order Taylor expansion

$$\begin{aligned} Q_f(y; x^{(t)}) \quad := \quad & f(x^{(t)}) + \nabla f(x^{(t)})^\top (y - x^{(t)}) \\ & + \frac{1}{2}(y - x^{(t)})^\top \nabla^2 f(x^{(t)})(y - x^{(t)}), \end{aligned}$$

where $\nabla^2 f(x^{(t)})$ is the Hessian matrix. To generate the next iterate that decreases the objective while remaining feasible, the method minimizes the quadratic model $Q_f(y; x^{(t)})$ over the original feasible set Ω:

$$\tilde{x}^{(t)} = \arg\min_{y \in \Omega} Q_f(y; x^{(t)}).$$

The new iterate is then obtained by simply setting

$$x^{(t+1)} = x^{(t)} + \beta(\tilde{x}^{(t)} - x^{(t)}),$$

where $\beta \in (0, 1)$ is the step-size selected via backtracking line search. If the backtracking line search always selects the step-size $\beta = 1$, then this method achieves a superlinear rate of convergence in the neighborhood of any point that satisfies the second-order sufficiency conditions for a minimizer [2, §2.3]. Such an appealing theoretical property of the constrained Newton method inspires us to adapt this method to the sparsity-constrained optimization problem (1).

2.2. The NTGP algorithm

NTGP is a greedy selection algorithm to approximately estimate the solution of (1). A high level summary of its procedure is described in Algorithm 1. The procedure generates a sequence of intermediate k-sparse vectors $x^{(0)}, x^{(1)}, \ldots$ from an initial sparse approximation $x^{(0)}$. At the time instance t, $x^{(t)}$ is selected as an approximate solution (up to a precision ϵ) to the quadratic model $Q_f(y; x^{(t-1)})$ over the constraint $\|y\|_0 \le k$. It will be assumed throughout the paper that the cardinality k is known; in practice this quantity may be regarded as a tunable parameter of the algorithm which can be selected via cross-validation.

Algorithm 1: Newton Greedy Pursuit (NTGP).

```
Initialization: $x^{(0)}$ with $\|x^{(0)}\|_0 \leq k$.
for $t = 1, 2, ...$ do
    Find any $x^{(t)}$ with $\|x^{(t)}\|_0 \leq k$ such that for all $\bar{y}$ with $\|\bar{y}\|_0 \leq k$,
        $Q_f(x^{(t)}; x^{(t-1)}) \leq Q_f(\bar{y}; x^{(t-1)}) + \epsilon,$    (3)
    where $\epsilon \geq 0$ controls the solution precision.
end
Output: $x^{(t)}$.
```

2.3. Sparse recovery analysis

We require some technical conditions under which the accuracy of NTGP can be guaranteed. We first introduce the following concept of stable restricted Hessian [1] which characterizes the curvature of the cost function over sparse subspaces.

Definition 1 (Stable Restricted Hessian). *Suppose that f is twice continuously differentiable. For all s-sparse vectors x, let*

$$M_s(x) = \sup_u \left\{ u^\top \nabla^2 f(x) u \mid \|u\|_0 \leq s, \|u\|_2 = 1 \right\}$$

and

$$m_s(x) = \inf_u \left\{ u^\top \nabla^2 f(x) u \mid \|u\|_0 \leq s, \|u\|_2 = 1 \right\}.$$

Then we say f has a Stable Restricted Hessian (SRH) with constant ρ_s, or in short ρ_s-SRH, if $1 \leq M_s(x)/m_s(x) \leq \rho_s$ holds for all s-sparse x.

The SRH property is analogue to the restricted isometry property [5] in standard CS analysis. It basically requires that the curvature of the cost function over the sparse subspaces can be bounded locally from above and below such that the corresponding bounds have the same order.

We next introduce the concept of restricted Lipschitz Hessian which characterizes the continuity of the Hessian matrix over sparse subspaces. To simplify the notation, we will abbreviate in the following analysis $\nabla_F f := (\nabla f)_F$, $\nabla_s f := (\nabla f)_s$ and $\nabla_F^2 f := (\nabla^2 f)_F$.

Definition 2 (Restricted Lipschitz Hessian). *Suppose that f is twice continuously differentiable. We say f has Restricted Lipschitz Hessian with constant γ_s (or γ_s-RLH), if*

$$\|\nabla_F^2 f(x) - \nabla_F^2 f(y)\|_2 \leq \gamma_s \|x - y\|_2$$

for all index set F with cardinality $|F| \leq s$ and all x, y with $supp(x) \cup supp(y) \subseteq F$.

The following is our main result on the convergence and parameter estimation error of NTGP.

Theorem 1. *Suppose that f is a twice continuously differentiable function that has ρ_s-SRH. Let $\bar{x}$ be a $\bar{k}$-sparse vector and $k \geq \bar{k}$. Let $s = 2k + \bar{k}$.*

(a) Suppose that for some $m_s > 0$ we have $m_s(x) \geq m_s$ for all s-sparse vector x. Then it holds that

$$\|x^{(t)} - \bar{x}\|_2 \leq o(\|x^{(t-1)} - \bar{x}\|_2) + \delta_s(\bar{x}, \epsilon), \quad (4)$$

where $\delta_s(\bar{x}, \epsilon) := m_s^{-1}(1 + \rho_s^{1/2})\|\nabla_s f(\bar{x})\|_2 + m_s^{-1/2}\epsilon^{1/2}$.

(b) Furthermore if f has γ_s-RLH, then

$$\|x^{(t)} - \bar{x}\|_2 \leq \theta_s \|x^{(t-1)} - \bar{x}\|_2^2 + \delta_s(\bar{x}, \epsilon), \quad (5)$$

where $\theta_s := 0.5\gamma_s m_s^{-1}(1 + \rho_s^{1/2})$.

A proof of this theorem is provided in Appendix A.1.

Remark 1. *The main message Theorem 1 conveys is that under proper conditions, up to a precision $\delta_s(\bar{x}, \epsilon)$, the sequence $\{x^{(t)}\}$ generated by NTGP locally converges superlinearly towards any $\bar{k}$-sparse vector $\bar{x}$. If we further assume that f has γ_s-RLH, then the part (b) of Theorem 1 shows that the rate of convergence is at least quadratic. The estimation error term $\delta_s(\bar{x}, \epsilon)$ indicates how accurate the estimate can be. It is controlled by the norm $\|\nabla_s f(\bar{x})\|_2$ and the precision ϵ for minimizing the ℓ_0-constrained quadratic model. Particularly, if $\nabla f(\bar{x}) = 0$ (i.e. $\bar{x}$ is the station point of (1)) and $\epsilon = 0$, then $\delta_s(\bar{x}, \epsilon) = 0$. In this case, provided that $x^{(0)}$ is close enough to $\bar{x}$, NTGP is able to exactly recover $\bar{x}$ asymptotically at superlinear rate. This result is analogous to accuracy guarantees for estimation from noisy measurements in CS [5, 12], but with improved order of convergence rate.*

Remark 2. *The local superlinear rate of convergence contradicts NTGP from those first-order greedy selection methods which converge sublinearly/linearly. The two key conditions used in our analysis are: (i) f is twice continuously differentiable; and (ii) f has SRH. These two conditions are also key to the analysis of GraSP [1] as a gradient support pursuit method with linear rate of convergence. By using a slightly stronger assumption that f has RLH, we further derived the local quadratic rate of convergence for NTGP.*

The following result is a direct consequence of the part (b) in Theorem 1. This result more precisely shows the conditions under which the local quadratic rate of convergence and the estimation error bound can be guaranteed. Its proof can be found in Appendix A.2.

Corollary 1. *Under the assumptions in Theorem 1(b), if $\|x^{(0)} - \bar{x}\|_2 \leq \theta_s^{-1}/4$ and $\delta_s(\bar{x}, \epsilon) \leq \theta_s^{-1}/8$, then we have*

$$\|x^{(t)} - \bar{x}\|_2 \leq \theta_s^{-1} \left(1/4 - 2\theta_s \delta_s(\bar{x}, \epsilon)\right)^{2^t} + 2\delta_s(\bar{x}, \epsilon). \quad (6)$$

2.4. Constrained quadratic model

At the t-th iteration, NTGP needs to approximately minimize the following sparsity-constrained quadratic program:

$$\min_y Q_f(y; x^{(t-1)}), \quad \text{s.t. } \|y\|_0 \leq k. \tag{7}$$

This reduced problem is a standard CS problem which is still NP-hard. We resort to the IHT [3] method, as described in Algorithm 2, to approximately solve such an inner loop subproblem.

Algorithm 2: Iterative Hard Thresholding (IHT) for solving the subproblem (7).

```
Initialization: $y^{(0)} = x^{(t-1)}$.
for $\tau = 1, 2, ...$ do
    (S1) Compute gradient descent:
    $\tilde{y}^{(\tau)} = y^{(\tau-1)} - \eta \nabla Q_f(y^{(\tau-1)}; x^{(t-1)})$;
    (S2) Identify support: $T^{(\tau)} = \text{supp}(\tilde{y}^{(\tau)}, k)$;
    (S3) Minimizer over support:
    $y^{(\tau)} = \arg\min_{\text{supp}(y) \subseteq T^{(\tau)}} Q_f(y; x^{(t-1)})$;
end
```

The following result establishes the convergence property of $\{y^{(\tau)}\}$ generated by Algorithm 2.

Theorem 2. *Suppose that f is a twice continuously differentiable function. Let $\bar{y}$ be a $\bar{k}$-sparse vector. Given that $\eta \in (0, 2/M_s(x^{(t-1)}))$ and $k \geq \bar{k}\left(1 + \frac{4M_s(x^{(t-1)})}{m_s^2(x^{(t-1)})(2\eta - \eta^2 M_s(x^{(t-1)}))}\right)$, if $Q_f(y^{(\tau)}) \geq Q_f(\bar{y})$, then we have*

$$Q_f(y^{(\tau)}; x^{(t-1)}) \leq Q_f(\bar{y}; x^{(t-1)}) + (1-\nu)^\tau \triangle_0,$$

where $\nu = \frac{(2\eta - \eta^2 M_s(x^{(t-1)}))m_s(x^{(t-1)})}{2\bar{k}}$ and $\triangle_0 = Q_f(y^{(0)}; x^{(t-1)}) - Q_f(\bar{y}; x^{(t-1)})$.

A proof of this theorem is given in Appendix A.3.

Remark 3. *One implication of Theorem 2 is that if the step size is set to be $\eta = 1/M_s(x^{(t-1)})$ and f has ρ_s-SRH, then for any $\bar{k}$-sparse $\bar{y}$, after at most $\tau = (\ln\epsilon - \ln\triangle_0)/\ln(1 - (2\bar{k}\rho_s)^{-1})$ steps of iteration we will have $Q_f(y^{(\tau)}; x^{(t-1)}) \leq Q_f(\bar{y}; x^{(t-1)}) + \epsilon$. This is desired at each iteration of Algorithm 1. Note that $M_s(x^{(t-1)})$ is the so called* s-sparse eigenvalue *of the Hessian $\nabla^2 f(x^{(t-1)})$ [17]. In our implementation, we choose $s = 3k$ and estimate $M_s(x^{(t-1)})$ using the truncated power (TPower) method [17] which is quite efficient and accurate in our practice. We then set η as the inverse of $M_s(x^{(t-1)})$.*

Computational cost. The overhead cost at each iteration of Algorithm 2 is dominated by the steps S1 and S3. To evaluate ∇Q_f in the step S1, we only need to calculate product of the Hessian $\nabla^2 f(x^{(t-1)})$ with the $2k$-sparse vector $y^{(\tau-1)} - x^{(t-1)}$. The computational complexity of this step is $O(kp)$. In the step S3, $y^{(\tau)}$ is given by the solution of the following linear system

$$\nabla^2_{T^{(\tau)}} f^{(t-1)} y = -\nabla_{T^{(\tau)}} f^{(t-1)} + [\nabla^2 f^{(t-1)} x^{(t-1)}]_{T^{(\tau)}},$$

in which we have used the abbreviation $f^{(t-1)} := f(x^{(t-1)})$. Therefore, to estimate $y^{(\tau)}$ it is sufficient to compute the principle submatrix $\nabla^2_{T^{(\tau)}} f^{(t-1)}$ restricted on the index set $T^{(\tau)}$ and then solve a linear system. Since $|T^{(\tau)}| \leq 3k$, a direct solution leads to $O(k^3)$ complexity[1]. When k is relatively large, the step S3 can be solved via general purpose convex solvers such as PQN used in our implementation. One may compare the per-iteration complexity of NTGP with that of GraSP [1]: the former minimizes a quadratic function over a sparsity constraint whilst the latter minimizes an arbitrary convex objective over a fixed supporting set. It is open to compare the complexity of these two subproblems for general cases. As we will see in §3 that NTGP is as efficient as GraSP in the considered sparse logistic regression tasks.

2.5. Example: sparse logistic regression

As an example, we specialize NTGP to the sparse logistic regression problem which is one of the most popular models in pattern recognition and machine learning. Given a set of n independently drawn data samples $\{(u_i, v_i)\}_{i=1}^n$ where $u_i \in \mathbb{R}^p$ is the feature and $v_i \in \{-1, +1\}$ is the binary label, logistic regression learns parameters w so as to minimize the logistic loss

$$l(w) := \frac{1}{n}\sum_{i=1}^n \log(1 + \exp(-2v_i w^\top u_i)).$$

The following sparsity-constrained ℓ_2-regularized logistic regression is widely used in high-dimensional analysis [1]:

$$\min_w f(w) = l(w) + \frac{\lambda}{2}\|w\|_2^2, \quad \text{s.t. } \|w\|_0 \leq k, \tag{8}$$

where $\lambda > 0$ is the regularization strength parameter. Obvious $f(w)$ is λ-strongly convex, thus it has a unique minimum. The cardinality constraint enforces sparse solution.

Verifying SRH and RLH. Let $\sigma(z) = 1/(1 + \exp(-z))$ be the sigmoid function. It is easy to show that the gradient $\nabla f(w) = Ua(w)/n + \lambda w$ where $a(w) \in \mathbb{R}^n$ with $[a(w)]_i = -2v_i(1 - \sigma(2v_i w^\top u_i))$; and the Hessian $\nabla^2 f(w) = U\Lambda(w)U^\top/n + \lambda I$ where $\Lambda(w)$ is an $n \times n$ diagonal matrix whose diagonal entries $[\Lambda(w)]_{ii} = 4\sigma(2v_i w^\top u_i)(1 - \sigma(2v_i w^\top u_i))$.

The continuity of $\sigma(z)$ implies the continuity of $\nabla^2 f(w)$, i.e., $f(w)$ is twice continuously differentiable. It

[1] We consider here that solving linear systems takes cubic time. This time complexity however can be improved.

has been shown in [1, Corollary 1] that under mild conditions, $f(w)$ has SRH with overwhelming probability. The following result further verifies that f has RLH.

Proposition 1. *Assume that for any index set F with $|F| \leq s$ we have $\forall i$, $\|(u_i)_F\|_2 \leq R_s$. Then the ℓ_2-regularized logistic loss has γ_s-RLH with $\gamma_s \leq 24sR_s^3$.*

A proof of this proposition is provided in Appendix A.4. Concerning the estimation error bound $\delta_s(\bar{x}, \epsilon)$, if the precision $\epsilon = 0$, then $\delta_s(\bar{x}, \epsilon)$ is proportional to $\|\nabla_s f(w)\|_2$. The bounding analysis of $\|\nabla_s f(w)\|_2$ for the ℓ_2-regularized logistic loss can be found in [1].

3. Experimental Results

In this section, we show some numerical results to demonstrate the effectiveness and efficiency of NTGP when applied to sparse logistic regression tasks. All the considered algorithms are implemented in Matlab 7.12 running on a person desktop with Core i7 CPU@3.40GHz.

3.1. Simulation study

In our simulations, we consider a data model with sparse parameter $\bar{w}$ is a $p = 2000$ dimensional vector that has $\bar{k} = 200$ nonzero entries drawn independently from the standard Gaussian distribution. Each data sample u is a normally distributed vector. The data labels, $v \in \{-1, 1\}$, are then generated randomly according to the Bernoulli distribution

$$\mathbb{P}(v = 1|u; \bar{w}) = \frac{\exp(2\bar{w}^\top u)}{1 + \exp(2\bar{w}^\top u)}.$$

We fix the sparsity parameter $k = \bar{k}$ and the regularization parameter $\lambda = 10^{-4}$ in (8). We compare NTGP with three first-order greedy selection methods: the GraSP [1], the FCFGS [14] and the FoBa [18]. All these three methods are designed to solve the sparsity-constrained optimization problem (1). For each choice of the sample size n with $n/p \in \{0.1, 0.15, 0.2, ..., 1\}$ we generate 100 independent copies of data and the associated labels. For each copy, we generate an independent copy of the same size for validating the tuning parameter k. We initialize $w^{(0)} = 0$ for the considered algorithms. Throughout our experiment, we set the stopping criterion of $|f(w^{(t)}) - f(w^{(t-1)})|/|f(w^{(t-1)})| \leq 10^{-4}$.

Figure 3 shows the average values of the estimation error (i.e., $\|\hat{w} - \bar{w}\|_2$), the support recovery precision and the CPU running time achieved by the considered algorithms under a wide range of sampling ratio n/p. The following observations are made from this figure: 1) in terms of parameter estimation performance, NTGP is consistently better than the other three considered methods; 2) in terms of support recovery performance, NTGP significantly outperforms the other three considered methods; 3) in terms of running efficiency, NTGP is as efficient as GraSP and these two are significantly faster than FBS and FoBa. Overall, for this simulation study, we conclude that NTGP is able to find much more accurate solutions than the three considered first-order greedy selection methods, without sacrificing efficiency.

3.2. Real data

The considered algorithms are also assessed on the rcv1.binary dataset (p = 47,236) which is a standard benchmark for binary classification on sparse data [10]. A training subset of size n = 20,242 and a testing subset of size 20,000 are used. We set the regularization parameter $\lambda = 10^{-5}$ and test with sparsity parameter k ranging from 500 to 5000. The initial vector is set to be $w^{(0)} = 0$ for all the considered algorithms.

The left two panels of Figure 2 plot the curves of empirical logistic loss verses number of function/gradient evaluations, for $k = 1000$ and $k = 5000$ respectively. It can be seen from these two plots that NTGP converges faster than the other three algorithms. This confirms the theoretical prediction of Theorem 1. The classification errors of the considered algorithms are shown in the right most panel of Figure 2. It can be seen that NTGP is comparable to GraSP in testing performance and they are both superior to FCFGS and FoBa. Also, we observe that the testing performances of FCFGS and FoBa are insensitive to k. This is because these two methods pursuit support in a sequential way.

4. Conclusions

We proposed NTGP as a quadratic approximation greedy selection method for sparsity-constrained optimization problem. The main idea is to construct a second-order Taylor expansion to approximate the objective function and solve the resultant sparsity-constrained quadratic model at each iteration. Theoretically we showed that up to an estimation error, NTGP converges superlinearly in a vicinity of a target sparse solution. The estimation error is controlled by the gradient norm at the target sparse solution, as well as the solution quality of the constrained quadratic subproblem. We demonstrated the performance of NTGP when applied to sparse logistic regression tasks. To conclude, NTGP is a theoretically and computationally sound second-order greedy support pursuit method.

Acknowledgement

We would like to thank the three anonymous reviewers for their constructive comments on this paper. Xiao-Tong Yuan is supported by NUIST Faculty Startup Grant No.S8113029001 and NSFC No.61300162, Qingshan Liu is supported by NSFC No.61272223 and NSF of Jiangsu Province No.BK2012045.

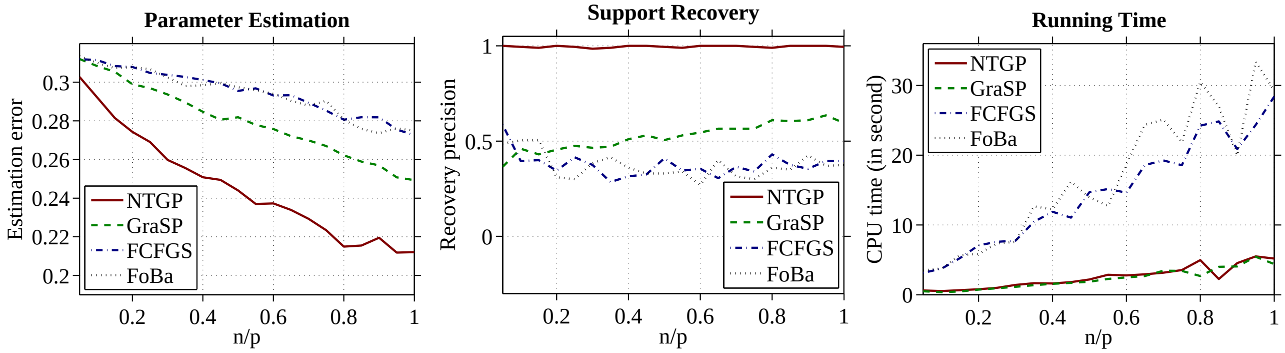

Figure 1. Simulated data: parameter estimation error (left panel), support recovery precision (middle panel) and running time (right panel).

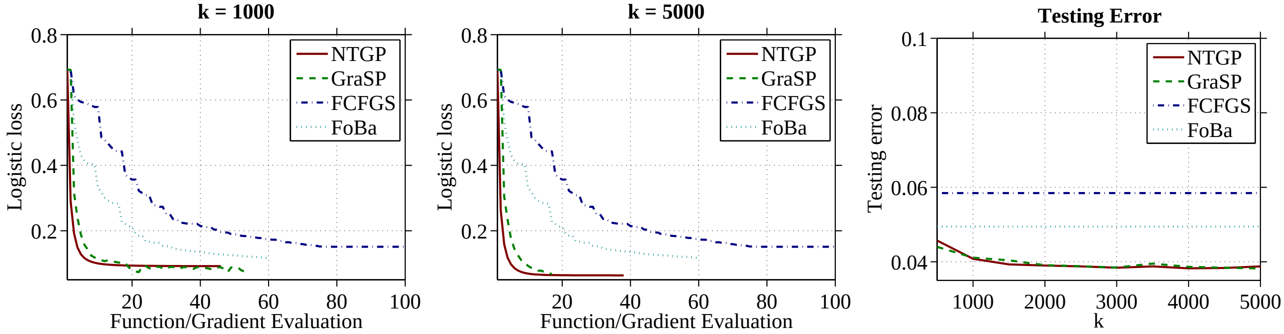

Figure 2. rcv1.binary data: objective value convergence curves (left and middle panels) and testing error curves (right panel).

A. Technical Proofs

This appendix section is devoted to proving the theoretical results stated in this paper.

A.1. Proof of Theorem 1

We need the following lemma which shows the progress of Newton iteration carried out in a sparse subspace spanned over an index set F.

Lemma 1. *Assume that f is twice continuously differentiable. Consider an index set F with $|F| \leq s$ and an vector x satisfies supp$(x) \subseteq F$. Assume that $\nabla_F^2 f(x)$ is invertible. Let $y = x - [\nabla_F^2 f(x)]^{-1}\nabla_F f(x)$. Consider any $\bar{x}$ with supp$(\bar{x}) \cup$ supp$(x) \subseteq F$.*

(a) If $m_s(x) \geq m_s > 0$ for all s-sparse vector x, then we have

$$\|y - \bar{x}\|_2 \leq o(\|x - \bar{x}\|_2) + m_s^{-1}\|\nabla_s f(\bar{x})\|_2.$$

(b) Furthermore if f has γ_s-RLH, then we have

$$\|y - \bar{x}\|_2 \leq 0.5\gamma_s m_s^{-1}\|x - \bar{x}\|_2^2 + m_s^{-1}\|\nabla_s f(\bar{x})\|_2.$$

Proof. **Part (a):** From the definition of y we have that

$$\begin{aligned} & y - \bar{x} = x - \bar{x} - [\nabla_F^2 f(x)]^{-1}\nabla_F f(x) \\ = & [\nabla_F^2 f(x)]^{-1}[\nabla_F^2 f(x)(x - \bar{x}) - \nabla_F f(x) + \nabla_F f(\bar{x})] \\ & -[\nabla_F^2 f(x)]^{-1}\nabla_F f(\bar{x}). \end{aligned} \tag{9}$$

Since Taylor's theorem tells us that

$$\nabla_F f(x) - \nabla_F f(\bar{x}) = \int_0^1 \nabla_F^2 f(x + t(\bar{x} - x))(x - \bar{x})dt,$$

we have that

$$\begin{aligned} & \|\nabla_F^2 f(x)(x - \bar{x}) - \nabla_F f(x) + \nabla_F f(\bar{x})\|_2 \\ = & \left\| \int_0^1 [\nabla_F^2 f(x) - \nabla_F^2 f(x + t(\bar{x} - x))](x - \bar{x})dt \right\|_2 \\ \leq & \int_0^1 \|\nabla_F^2 f(x) - \nabla_F^2 f(x + t(\bar{x} - x))\|_2 dt \|x - \bar{x}\|_2. \end{aligned}$$

By substituting the preceding inequality to (9) we obtain

$$\begin{aligned} \|y - \bar{x}\|_2 \leq & \; m_s^{-1}\|x - \bar{x}\|_2 \times \\ & \int_0^1 \|\nabla_F^2 f(x) - \nabla_F^2 f(x + t(\bar{x} - x))\|_2 dt \\ & + m_s^{-1}\|\nabla_s f(\bar{x})\|_2. \end{aligned}$$

Since f is twice continuously differentiable, we have that $\nabla_F^2 f$ is continuous. It follows that $\|y - \bar{x}\|_2 \leq o(\|x - \bar{x}\|_2) + m_s^{-1}\|\nabla_s f(\bar{x})\|_2$.

Part(b): If f has γ_s-RLH, then the proceeding relation yields

$$\begin{aligned} \|y - \bar{x}\|_2 \leq & \; m_s^{-1}\|x - \bar{x}\|_2^2 \int_0^1 \gamma_s t dt + m_s^{-1}\|\nabla_s f(\bar{x})\|_2 \\ = & \; 0.5\gamma_s m_s^{-1}\|x - \bar{x}\|_2^2 + m_s^{-1}\|\nabla_s f(\bar{x})\|_2. \end{aligned}$$

This proves the desired result. □

We are now in the position to prove Theorem 1.

Proof of Theorem 1. **Part (a):** Let $F_t := \mathrm{supp}(x^{(t)})$ and $F := F_{t-1} \cup F_t \cup \mathrm{supp}(\bar{x})$. Obviously $|F| \leq s$. Consider the following vector

$$x'_t = x^{(t-1)} - [\nabla^2_F f(x^{(t-1)})]^{-1} \nabla_F f(x^{(t-1)}).$$

It is easy to verify that for any x with $\mathrm{supp}(x) \subseteq F$

$$Q_f(y; x^{(t-1)}) = (y - x'_t)^\top \nabla^2_F f(x^{(t-1)})(y - x'_t) + const. \tag{10}$$

By using the part (a) in Lemma 1, we have that

$$\|x'_t - \bar{x}\|_2 \leq o(\|x^{(t-1)} - \bar{x}\|_2) + m_s^{-1} \|\nabla_s f(\bar{x})\|_2. \tag{11}$$

It follows from triangle inequality and (10) that

$$\begin{aligned}
& \|x^{(t)} - \bar{x}\|_2 \\
\leq\ & \|x^{(t)} - x'_t\|_2 + \|x'_t - \bar{x}\|_2 \\
\leq\ & m_s(x^{(t-1)})^{-1/2} \\
& \times \sqrt{(x^{(t)} - x'_t)^\top \nabla^2_F f(x^{(t-1)})(x^{(t)} - x'_t)} \\
& + \|x'_t - \bar{x}\|_2 \\
\leq\ & m_s(x^{(t-1)})^{-1/2} \\
& \times \sqrt{(\bar{x} - x'_t)^\top \nabla^2_F f(x^{(t-1)})(\bar{x} - x'_t) + \epsilon} \\
& + \|x'_t - \bar{x}\|_2 \\
\leq\ & m_s(x^{(t-1)})^{-1/2} \sqrt{(\bar{x} - x'_t)^\top \nabla^2_F f(x^{(t-1)})(\bar{x} - x'_t)} \\
& + m_s(x^{(t-1)})^{-1/2} \epsilon^{1/2} + \|x'_t - \bar{x}\|_2 \\
\leq\ & ((M_s(x^{(t-1)})/m_s(x^{(t-1)}))^{1/2} + 1)\|x'_t - \bar{x}\|_2 \\
& + m_s(x^{(t-1)})^{-1/2} \epsilon^{1/2} \\
\leq\ & (\rho_s^{1/2} + 1)\|x'_t - \bar{x}\|_2 + m_s^{-1/2} \epsilon^{1/2},
\end{aligned} \tag{12}$$

where the second "$\leq$" follows (3) and (10). By combing (11) and (12) we obtain

$$\begin{aligned}
& \|x^{(t)} - \bar{x}\|_2 \\
\leq\ & o(\|x^{(t-1)} - \bar{x}\|_2) + m_s^{-1}(1 + \rho_s^{1/2})\|\nabla_s f(\bar{x})\|_2 \\
& + m_s^{-1/2} \epsilon^{1/2}.
\end{aligned}$$

This proves (4).

Part (b): By using the part (b) in Lemma 1, we obtain

$$\begin{aligned}
\|x'_t - \bar{x}\|_2 \leq\ & 0.5 \gamma_s m_s^{-1} \|x^{(t-1)} - \bar{x}\|_2^2 \\
& + m_s^{-1} \|\nabla_s f(\bar{x})\|_2.
\end{aligned} \tag{13}$$

By combing (13) and (12) we obtain

$$\begin{aligned}
\|x^{(t)} - \bar{x}\|_2 \leq\ & 0.5 \gamma_s m_s^{-1}(1 + \rho_s^{1/2})\|x^{(t-1)} - \bar{x}\|_2^2 \\
& + m_s^{-1}(1 + \rho_s^{1/2})\|\nabla_s f(\bar{x})\|_2 \\
& + m_s^{-1/2} \epsilon^{1/2}.
\end{aligned}$$

This proves (5). □

A.2. Proof of Corollary 1

Proof. This can be proved by induction. Obviously, inequality (6) holds for $t = 0$. Assume that (6) holds for $t - 1$. From the part (b) of Theorem 1 we obtain

$$\begin{aligned}
\|x^{(t)} - \bar{x}\|_2 \leq\ & \theta_s \|x^{(t-1)} - \bar{x}\|_2^2 + \delta_s(\bar{x}, \epsilon) \\
\leq\ & \theta_s [\theta_s^{-1}(1/4 - 2\theta_s \delta_s(\bar{x}, \epsilon))^{2^{t-1}} \\
& + 2\delta_s(\bar{x}, \epsilon)]^2 + \delta_s(\bar{x}, \epsilon) \\
\leq\ & \theta_s^{-1}(1/4 - 2\theta_s \delta_s(\bar{x}, \epsilon))^{2^t} \\
& + 2\delta_s(\bar{x}, \epsilon),
\end{aligned}$$

where the second "$\leq$" follows from the assumption $\delta_s(\bar{x}, \epsilon) \leq \theta_s^{-1}/8$. This finishes the induction. □

A.3. Proof of Theorem 2

Proof. Let $F = T^{(\tau-1)} \cup \mathrm{supp}(\bar{y})$. Note that if $|F - \mathrm{supp}(\bar{y})| < \bar{k}$, then we add additional (arbitrary) indices to F so that $|F - \mathrm{supp}(\bar{y})| = \bar{k}$. For the sake of notation simplicity, we write $Q_f(y; x^{(t-1)})$ as $Q_f(y)$, $\nabla Q_f(y^{(\tau)})$ as $\nabla Q_f^{(\tau)}$ and $m_s(x^{(t-1)})$ (and $M_s(x^{(t-1)})$) as $m_s^{(t-1)}$ (and $M_s^{(t-1)}$). From Definition 1 we have

$$\begin{aligned}
& \frac{m_s^{(t-1)}}{2} \|\bar{y} - y^{(\tau-1)}\|_2^2 \\
\leq\ & Q_f(\bar{y}) - Q_f(y^{(\tau-1)}) - (\bar{y} - y^{(\tau-1)})^\top \nabla Q_f^{(\tau-1)} \\
\leq\ & Q_f(\bar{y}) - Q_f(y^{(\tau-1)}) + \frac{m_s^{(t-1)}}{4} \|\bar{y} - y^{(\tau-1)}\|_2^2 \\
& + \frac{1}{m_s^{(t-1)}} \|\nabla_F Q_f^{(\tau-1)}\|_2^2,
\end{aligned}$$

where the last inequality follows from Cauchy-Schwartz inequality and $a^2/(4m) + mb^2 \geq ab$. From the step S3 we know that $\nabla_{T^{(\tau-1)}} Q_f^{(\tau-1)} = 0$. By definition of $T^{(\tau)}$ we may decompose $T^{(\tau)} = G_1 \cup (T^{(\tau-1)} - G_2)$ with $G_1 \subseteq \mathrm{supp}(\nabla Q_f^{(\tau-1)})$, $G_2 \subseteq T^{(\tau-1)}$ and $|G_1| = |G_2| = k' \leq k$. That is, G_1 contains the top k' (in magnitude) entries in $\nabla Q_f^{(\tau-1)}$ while G_2 contains the bottom k' entries in $y^{(\tau-1)}$. We assume without loss of generality that $k' \geq 1$ (otherwise the algorithm terminates). Combing these facts and the preceding inequality we get

$$\begin{aligned}
& (\bar{k}/k') \|\nabla_{G_1} Q_f^{(\tau-1)}\|_2^2 \geq \|\nabla_F Q_f^{(\tau-1)}\|_2^2 \\
\geq\ & m_s^{(t-1)} \left[\delta Q_f + \frac{m_s^{(t-1)}}{4} \|\bar{y} - y^{(\tau-1)}\|_2^2 \right] \\
\geq\ & m_s^{(t-1)} \left[\delta Q_f + \frac{(k - \bar{k}) m_s^{(t-1)}}{4k'} \|y_{G_2}^{(\tau-1)}\|_2^2 \right],
\end{aligned} \tag{14}$$

where $\delta Q_f = Q_f(y^{(\tau-1)}) - Q_f(\bar{y})$. Now let $z^{(\tau)} := y_{T^{(\tau)}}^{(\tau)} = y^{(\tau-1)} + \Delta^{(\tau-1)}$ where

$$\Delta^{(\tau-1)} = -\eta \nabla_{G_1} Q_f^{(\tau-1)} - y_{G_2}^{(\tau-1)}.$$

From steps S1 and S3 in Algorithm 2 we have that

$$\begin{aligned}
& Q_f(y^{(\tau)}) \leq Q_f(z^{(\tau)}) \\
\leq\ & Q_f(y^{(\tau-1)}) + \langle \nabla Q_f^{(\tau-1)}, \Delta^{(\tau-1)} \rangle \\
& + \frac{M_s^{(t-1)}}{2} \|\Delta^{(\tau-1)}\|_2^2 \\
\leq\ & Q_f(y^{(\tau-1)}) + \frac{M_s^{(t-1)}}{2} \|y_{G_2}^{(\tau-1)}\|_2^2 \\
& - \frac{2\eta - \eta^2 M_s^{(t-1)}}{2} \|\nabla_{G_1} Q_f^{(\tau-1)}\|_2^2 \\
\leq\ & Q_f(y^{(\tau-1)}) - \frac{k'(2\eta - \eta^2 M_s^{(t-1)}) m_s^{(t-1)}}{2\bar{k}} \\
& (Q_f(y^{(\tau-1)}) - Q_f(\bar{y})) \\
& + \left(\frac{M_s^{(t-1)}}{2} - \frac{(k-\bar{k})(2\eta - \eta^2 M_s^{(t-1)})(m_s^{(t-1)})^2}{8\bar{k}} \right) \\
& \|y_{G_2}^{(\tau-1)}\|_{Frob}^2 \\
\leq\ & Q_f(y^{(\tau-1)}) \\
& - \frac{(2\eta - \eta^2 M_s^{(t-1)}) m_s^{(t-1)}}{2\bar{k}} (Q_f(y^{(\tau-1)}) - Q_f(\bar{y}))
\end{aligned}$$

where the third inequality follows (14), the last inequality follows $k \geq \bar{k} \left(1 + \frac{4M_s^{(t-1)}}{(m_s^{(t-1)})^2 (2\eta - \eta^2 M_s^{(t-1)})}\right)$ and $k' \geq 1$. This proves the desired result. □

A.4. Proof of Proposition 1

Proof. Consider an index set F with cardinality $|F| \leq s$ and all w, w' with $\text{supp}(w) \cup \text{supp}(w') \subseteq F$. Since $\sigma(z)$ is Lipschitz continuous with constant 1, we have that

$$\begin{aligned}
& |\sigma(2v_i w^\top u_i) - \sigma(2v_i w'^\top u_i)| \leq |2(w-w')^\top v_i u_i| \\
\leq\ & 2\|(u_i)_F\|_2 \|w - w'\|_2 \leq 2R_s \|w - w'\|_2.
\end{aligned}$$

Using this above inequality and the fact that $\sigma(z) \leq 1$ we obtain

$$\begin{aligned}
& |\sigma(2v_i w^\top u_i)(1 - \sigma(2v_i w^\top u_i)) \\
& - \sigma(2v_i w'^\top u_i)(1 - \sigma(2v_i w'^\top u_i))| \\
\leq\ & 3|\sigma(2v_i w^\top u_i) - \sigma(2v_i w'^\top u_i)| \leq 6R_s \|w - w'\|_2.
\end{aligned}$$

This yields that $\|\Lambda(w) - \Lambda(w')\|_2 \leq 24R_s\|w - w'\|_2$. Therefore we have

$$\begin{aligned}
& \|\nabla_F f(w) - \nabla_F f(w')\|_2 \\
\leq\ & \frac{1}{n} \|U_{F\bullet}\|_2^2 \|\Lambda(w) - \Lambda(w')\|_2 \\
\leq\ & \frac{24}{n} \|U_{F\bullet}\|_2^2 R_s \|w - w'\|_2 \leq 24 s R_s^3 \|w - w'\|_2,
\end{aligned}$$

where the last "$\leq$" follows from $\|U_{F\bullet}\|_2 \leq \sqrt{sn} \max_i \|(u_i)_F\|_2 \leq \sqrt{sn} R_s$. This proves the desired result. □

References

[1] S. Bahmani, B. Raj, and P. Boufounos. Greedy sparsity-constrained optimization. *Journal of Machine Learning Research*, 14:807–841, 2013.

[2] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, 2nd edition, 1999.

[3] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[4] R. Byrd, P. Lu, and J. Nocedal. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Stat. Comput.*, 16(5):1190–1208, 1995.

[5] E. J. Candès, J. K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[6] W. Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Info. Theory*, 55(5):2230–2249, 2009.

[7] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[8] C.-J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. K. Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. In *NIPS*, 2011.

[9] A. Jalali, C. C. Johnson, and P. K. Ravikumar. On learning discrete graphical models using greedy methods. In *Nueral Information Processing Systems*, 2011.

[10] D. Lewis, Y. Yang, T. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

[11] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995.

[12] D. Needell and J. A. Tropp. Cosamp: iterative signal recovery from incomplete and inaccurate samples. *IEEE Trans. Info. Theory*, 26(3):301–321, 2009.

[13] M. Schmidt, E. Berg, M. Friedlander, and K. Murphy. Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm. In *AISTATS*, pages 456–463, 2009.

[14] S. Shalev-Shwartz, N. Srebro, and T. Zhang. Trading accuracy for sparsity in optimization problems with sparsity constraints. *SIAM Journal on Optimization*, 20:2807–2832, 2010.

[15] J. Tropp and A. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Info. Theory*, 53(12):4655–4666, 2007.

[16] X.-T. Yuan and S. Yan. Forward basis selection for pursuing sparse representations over a dictionary. *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 35(12):3025–3036, 2013.

[17] X.-T. Yuan and T. Zhang. Truncated power method for sparse eigenvalue problems. *Journal of Machine Learning Research*, 14:899–925, 2013.

[18] T. Zhang. Adative forward-backward greedy algorithm for sparse learning with linear models. In *NIPS*, 2008.