l_q MATRIX COMPLETION

Goran Marjanovic, Student Member, IEEE and Victor Solo, Fellow, IEEE

School of Electrical Engineering and Telecommunications
The University of New South Wales, Sydney, Australia

ABSTRACT

Rank minimization problems, which consist of finding a matrix of minimum rank subject to linear constraints, have been proposed in many areas of engineering and science. A specific problem is the matrix completion problem in which a low rank data matrix is recovered from incomplete samples of its entries by solving a rank penalized least squares problem. The rank penalty is in fact the l_0 *norm* of the matrix singular values. A convex relaxation of this penalty is the commonly used l_1 norm of the matrix singular values. In this paper we bridge the gap between these two penalties and propose a simple method for solving the l_q, $q \in (0,1)$, penalized least squares problem for matrix completion. We illustrate with simulations comparing our method to others in terms of solution quality.

Index Terms— Matrix completion, matrix rank minimization, sparse, l_q optimization.

1. INTRODUCTION

In many applications noisy measurements $\mathbf{Y}$ are made of a limited number of entries in a matrix of interest $\mathbf{X}$. Matrix completion problems deal with *completing* $\mathbf{X}$ based on the observed entries such that the resulting matrix satisfies specific properties [1,2]. Here we focus on recovering a low rank matrix from a given subset of its entries. This is a recurring problem in applications such as collaborative filtering [2], dimensionality reduction [3], and multiclass learning [4]. A very popular example is the *Netflix* competition [5], where the rows of $\mathbf{X}$ correspond to viewers and the columns to viewer movie ratings. In this case, out of the potential 8.6 billion entries only 1.2% are observed on average. Thus, the task is to complete $\mathbf{X}$ in order to predict the unrated movie ratings. It is assumed that the completion of $\mathbf{X}$ has a low rank structure since the movies could be grouped into a small number of *genres*.

In theory and practise nuclear norm minimization is known to produce low rank matrix solutions [5–9]. Just as the l_1 norm promotes sparsity in sparse signal reconstruction [10], the nuclear norm, defined as the l_1 norm of matrix singular values, should promote matrix low rank in matrix completion.

We assume noisy incomplete observations $\mathbf{Y}_{m\times n}$ of a matrix $\mathbf{X}_{m\times n}$ according to the model $\mathbf{Y}_{ij} = \mathbf{X}_{ij} + \epsilon_{ij}$ where $(i,j) \in \Omega \subset \{1,\dots,m\} \times \{1,\dots,n\}$ and the noise ϵ_{ij} is i.i.d. Gaussian. Then the l_1 matrix completion problem reconstructs $\mathbf{X}$ by the l_1-penalized least squares problem:

$$\min_{\mathbf{X}} \frac{1}{2} \sum_{(i,j)\in\Omega} (\mathbf{Y}_{ij} - \mathbf{X}_{ij})^2 + \lambda \|\mathbf{X}\|_1 \qquad (1)$$

where $\|\cdot\|_1$ denotes the nuclear norm and $\lambda > 0$ is a penalty parameter. The first term penalizes the error between the observed entries in $\mathbf{Y}$ and corresponding entries in $\mathbf{X}$, while the second acts as a rank surrogate and is supposed to penalize the rank of $\mathbf{X}$.

From experimental studies it was evidenced that the nuclear norm could outperform the rank penalized estimator (l_0 *norm*[1] of the singular values) in terms of prediction accuracy [5]. A natural analogy is drawn with model selection in linear regression comparing the best subset regression (l_0 regularization) with the LASSO (l_1 regularization) [10]. In situations with moderate sparsity the LASSO outperformed best selection in terms of prediction accuracy.

On the other hand, there has recently been a significant interest in the use of the l_q, $q \in (0,1)$ penalty in variable selection and sparse reconstruction [11, 12]. The penalty is the sum of the absolute value of its components raised to the power of q, and naturally, one expects that using it allows a less *biased* (and/or sparser) solution to be found than using the l_1 norm. This was evidenced in extensive computational studies [11, 13].

Motivated by the above, the purpose of this paper is to present a method for *solving* (1) with the more general l_q, $q \in (0,1)$, rank penalty instead of the nuclear norm. This will be referred to as the l_q *matrix completion problem*. The new method, which we call l_qPG (l_q Proximal Gradient) is based on the majorization-minimization (MM) technique [14] that replaces a difficult minimization problem by a sequence of easier ones.

The remainder of the paper is organized as follows. In section 2 we state the l_q matrix completion problem. In section 3 we describe the l_qPG method and section 4 contains

This work was partly supported by an Australian Postgraduate Research Award.

[1] it is a quasi norm rather than a norm

comparisons between l_qPG and two other well known methods in matrix completion. Finally, in section 5 conclusions are drawn.

Notation: $\mathbf{I} : \mathbb{R} \to \mathbb{R}$ denotes the indicator function where $\mathbf{I}(x > a) = 1$ if $x > a$ and 0 otherwise. For any $\mathbf{X}_{m\times n}$ and $\mathbf{Z}_{m\times n}$ the Frobenius inner product and norm are respectively denoted by $\langle \mathbf{X}, \mathbf{Z}\rangle_F \triangleq \mathbf{tr}(\mathbf{X}^T\mathbf{Z})$ and $\|\mathbf{X}\|_F \triangleq \sqrt{\langle \mathbf{X}, \mathbf{X}\rangle_F}$. Finally, $\mathbf{diag}(\mathbf{x})$ denotes the diagonal matrix with the vector $\mathbf{x}$ on its main diagonal.

2. PRELIMINARIES

We define the l_q matrix penalty $\forall q \in (0, 1)$: $\|\mathbf{X}\|_q \triangleq \sum_i \sigma_i^q$ where σ_i's are the singular values of $\mathbf{X}$. So the l_q matrix completion problem is:

$$\min_{\mathbf{X}} J(\mathbf{X}) \triangleq \frac{1}{2} \sum_{(i,j)\in\Omega} (\mathbf{Y}_{ij} - \mathbf{X}_{ij})^2 + \lambda\|\mathbf{X}\|_q \qquad (2)$$

where $\lambda > 0$. Throughout the remainder of the paper we let: $f(\mathbf{X}) \triangleq 1/2\sum_{(i,j)\in\Omega}(\mathbf{Y}_{ij} - \mathbf{X}_{ij})^2$, and the gradient: $\nabla f(\mathbf{X})$ is Lipschitz continuous with Lipschitz constant L_f.

3. THE ALGORITHM

In this section we introduce the MM-based l_qPG algorithm to iteratively reduce J in (2). Considering the following quadratic approximation of $J(\mathbf{X})$:

$$Q(\mathbf{Z}, \mathbf{X}) \triangleq f(\mathbf{X}) + \langle \nabla f(\mathbf{X}), \mathbf{Z} - \mathbf{X}\rangle_F + \frac{L}{2}\|\mathbf{Z} - \mathbf{X}\|_F^2 + \lambda\|\mathbf{Z}\|_q$$

with $L \geq L_f$, by using [14, Lemma 2.1]: $J(\mathbf{Z}) \leq Q(\mathbf{Z}, \mathbf{X})$, $\forall \mathbf{X}, \mathbf{Z}$. Thus, Q is used as the majorizer of J. We next define the point to set map:

$$F(\mathbf{X}) \triangleq \left\{\mathbf{X}' : \mathbf{X}' = \arg\min_{\mathbf{Z}} Q(\mathbf{Z}, \mathbf{X})\right\}$$

which can have more than one element as Q is not convex. Using it we state our algorithm:

The l_qPG Algorithm

Given $\mathbf{X}_0$, repeat $\forall k \geq 0$. Choose

(1) $\mathbf{X}_{k+1} \in F(\mathbf{X}_k)$ if $\mathbf{X}_k \notin F(\mathbf{X}_k)$

(2) $\mathbf{X}_{k+1} = \mathbf{X}_k$ otherwise

Given the current iterate $\mathbf{X}$, if $\mathbf{X}' \in F(\mathbf{X})$ then $\mathbf{X} \notin F(\mathbf{X})$ iff $Q(\mathbf{X}', \mathbf{X}) < Q(\mathbf{X}, \mathbf{X})$, which is easy to check. In any case, $J(\mathbf{X}') \leq Q(\mathbf{X}', \mathbf{X}) \leq Q(\mathbf{X}, \mathbf{X}) = J(\mathbf{X})$.

The rest of this section focuses on the elements in $F(\mathbf{X})$. Defining $\mu \triangleq \lambda/L$ and $\mathbf{G} \triangleq \mathbf{X} - L^{-1}\nabla f(\mathbf{X})$, by rearranging and removing constant terms

$$\arg\min_{\mathbf{Z}} Q(\mathbf{Z}, \mathbf{X}) = \arg\min_{\mathbf{Z}} \Phi(\mathbf{Z}) \triangleq \frac{1}{2}\|\mathbf{Z} - \mathbf{G}\|_F^2 + \mu\|\mathbf{Z}\|_q$$

It can be shown [15], that $\mathbf{G}$ and the minimizers of $\Phi(\mathbf{Z})$ have the same left and right singular vectors. So, supposing $\mathbf{G}$ has rank r, let $\mathbf{U}_{m\times r}$ and $\mathbf{V}_{n\times r}$ be its left and right singular matrices, and $\sigma_1, \ldots, \sigma_r$ be its positive singular values. Substituting $\mathbf{Z} = \mathbf{U}\mathbf{diag}(d_1, \ldots, d_r)\mathbf{V}^T$ in $\Phi(\mathbf{Z})$ where d_i's are the r singular values of $\mathbf{Z}$, by unitary invariance of the Frobenius norm

$$\Phi(\mathbf{Z}) = \sum_{i=1}^{r} \phi(d_i) : \ \phi(d_i) \triangleq \frac{1}{2}(d_i - \sigma_i)^2 + \mu d_i^q \qquad (3)$$

which is separable in d_i and Φ is thus minimized by minimizing each $\phi(d_i)$, $d_i \geq 0$. As a result

$$\mathbf{U}\mathbf{diag}(\tau(\sigma_1), \ldots, \tau(\sigma_r))\mathbf{V}^T \in F(\mathbf{X})$$

where $\tau(\sigma_i) \in \arg\min_{d_i\geq 0} \phi(d_i)$ for $i = 1, \ldots, r$. Then

$$\delta \triangleq [2\mu(1 - q)]^{\frac{1}{2-q}} \text{ and } h \triangleq \delta + \mu q\delta^{q-1} \qquad (4)$$

Theorem 1. For $i = 1, \ldots, r$:

(a)
$$\tau(\sigma_i) = \begin{cases} \hat{x}(\sigma_i) & \sigma_i > h \\ 0 & otherwise \end{cases} \qquad (5)$$

where for $\sigma_i > h$ we have:

$$\hat{x}(\sigma_i) = \sigma_i - \mu q\hat{x}(\sigma_i)^{q-1} \in (\delta, \sigma_i) \qquad (6)$$

We give an iterative procedure for obtaining $\hat{x}(\sigma_i)$:

(b) For $\sigma_i > h$ let $\rho(x) \triangleq \sigma_i - \mu q x^{q-1}$ and generate an iterative sequence by:

$$x_{k+1} = \rho(x_k), \ x_0 \in [\delta, \sigma_i] \qquad (7)$$

Then $x_k \to \hat{x}(\sigma_i)$ as $k \to \infty$.

A sketch of the proof is given in the appendix.

Remark 1. Theorem 1 and l_qPG are valid with minor adjustments for $q = 0$ and $q = 1$: details are in [15]. So for simulation purposes we allow for $q \in [0, 1]$ in l_qPG.

Remark 2. While completing [15] we became aware of [16] with an algorithm for solving (2). However, their algorithm is flawed because their minimization of ϕ is wrong. As $\sigma_i > 0$, [16] claims a minimizer of ϕ is $\max\{0, \sigma_i - \mu q\sigma_i^{q-1}\} = (\sigma_i - \mu q\sigma_i^{q-1})\mathbf{I}(\sigma_i > \sigma^c)$ where $\sigma^c = [\mu q]^{\frac{1}{2-q}}$ (see [16, (15)]). Assuming we used a single iteration of (7) with $x_0 = \sigma_i$ to obtain (6), we see that the claimed minimizer is wrong because $\sigma^c \neq h$, i.e. the thresholding is wrong. As a result, the algorithm in [16] has an incorrect update and by [5, Remark 1] will not, as claimed, be a solution to $\arg\min_{\mathbf{Z}} Q(\mathbf{Z}, \mathbf{X})$ when having the same left and right singular vectors as $\mathbf{G}$.

A *global* minimizer $\mathbf{X}$ of J satisfies: $\mathbf{X} \in F(\mathbf{X})$, details are in [15]. Thus, we state the following important theorem:

Theorem 2. Suppose $\{\mathbf{X}_k\}_{k\in\mathbb{N}}$ is an infinite sequence generated by l_qPG . Then for any cluster point $\mathbf{X}^*$ of the sequence, $\mathbf{X}^* \in F(\mathbf{X}^*)$.

The proof is given in [15].

4. SIMULATIONS

Here we compare l_qPG with state-of-the-art convex and non-convex matrix completion algorithms from [5]: SOFT-IMPUTE and HARD-IMPUTE respectively. The SOFT-IMPUTE algorithm is designed for solving the l_1 matrix completion problem (1) and is equivalent to ISTA [14] and FPC [9] with unity step lengths. HARD-IMPUTE is designed for solving the l_0 matrix completion problem, i.e. (1) with the nuclear norm replaced by the rank penalty.

The set Ω is uniformly random and we let $|\Omega|/(m \times n)$ denote the Sampling Ratio where $|\Omega|$ is the size of Ω. We generate $\mathbf{X}$ with $m = n = 100$ and rank $r' = 10$ as in [5, 9] i.e. $\mathbf{X} = \mathbf{M}\mathbf{N}^T$ where $\mathbf{M}_{m\times r'}$ and $\mathbf{N}_{n\times r'}$ are randomly generated matrices with standard normal Gaussian entries.

In obtaining an algorithm estimator $\hat{\mathbf{X}}$ all algorithms are initialized with $\mathbf{X}_0 = 0$, use warm starting [5, 9] and are stopped after an equal number of iterations ($> 2.5 \times 10^3$). All computations are done in matlab and the matlab svd function is used. In l_qPG, $L = L_f = 1$.

The signal to noise ratio and prediction error are both defined as in [5], i.e.

$$\mathbf{SNR} \triangleq \sqrt{\frac{var(\mathbf{X})}{var(\epsilon)}} \text{ and } \mathbf{E}_{pr} \triangleq \frac{\sum_{(i,j)\notin\Omega}(\mathbf{X}_{ij} - \hat{\mathbf{X}}_{ij})^2}{\sum_{(i,j)\notin\Omega}\mathbf{X}_{ij}^2}$$

As in [5], we use $\mathbf{E}_{pr}$ to measure the quality of $\hat{\mathbf{X}}$.

	Sampling Ratio (%)			
Algorithm	20	30	40	50
HARD-IMPUTE	0.9952	0.9863	0.9670	0.9528
SOFT-IMPUTE	0.9175	0.8186	0.6761	0.5668
l_qPG	0.9175	0.7953	0.6217	0.5011

Table 1. Minimum average $\mathbf{E}_{pr}$ for different algorithms. $\mathbf{SNR} = 1$. For l_qPG the optimal $q = 1,\ 0.90,\ 0.75,\ 0.80$ for the respective sampling ratios.

	Sampling Ratio (%)			
Algorithm	20	30	40	50
HARD-IMPUTE	0.5564	0.1974	0.0424	0.0273
SOFT-IMPUTE	0.5801	0.2794	0.1290	0.0738
l_qPG	0.3888	0.0801	0.0393	0.0265

Table 2. Minimum average $\mathbf{E}_{pr}$ for different algorithms. $\mathbf{SNR} = 5$. For l_qPG the optimal $q = 0.70,\ 0.26,\ 0.36,\ 0.31$ for the respective sampling ratios.

	Sampling Ratio (%)			
Algorithm	20	30	40	50
HARD-IMPUTE	74 / 16	67 / 11	75 / 10	78 / 10
SOFT-IMPUTE	16 / 29	14 / 38	19 / 45	23 / 31
l_qPG	16 / 10	10 / 10	10 / 10	10 / 10

Table 3. Average recovered rank corresponding to each $\mathbf{E}_{pr}$: in Table 1 / Table 2 for different algorithms.

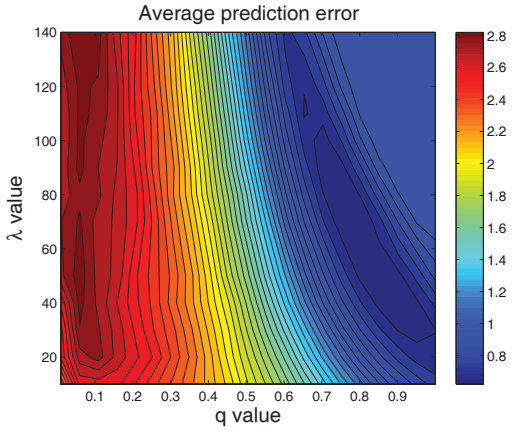

Fig. 1. Average $\mathbf{E}_{pr}$ for $\mathbf{SNR} = 1$ and Sampling Ratio 40% obtained with l_qPG. The optimal $(\lambda, q) = (80, 0.75)$ give the minimum average $\mathbf{E}_{pr} = 0.6217$ in table 1 for this scenario.

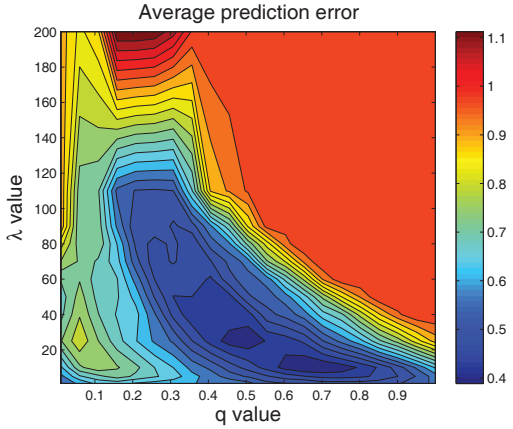

Fig. 2. Average $\mathbf{E}_{pr}$ for $\mathbf{SNR} = 5$ and Sampling Ratio 20% obtained with l_qPG. The optimal $(\lambda, q) = (10, 0.70)$ give the minimum average $\mathbf{E}_{pr} = 0.3888$ in table 2 for this scenario.

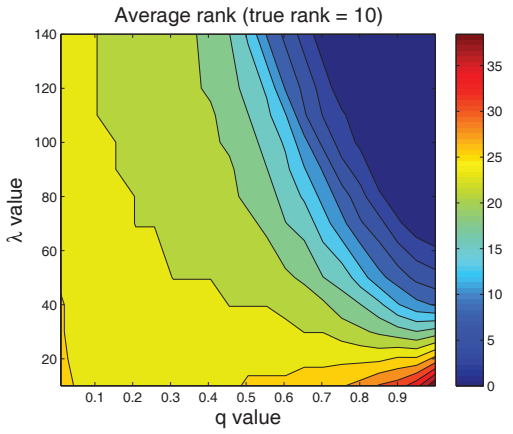

Fig. 3. Average recovered rank corresponding to fig.1. The optimal $(\lambda, q) = (80, 0.75)$ give the rank $(= 10)$ corresponding to the minimum average $\mathbf{E}_{pr}$ in fig.1 and given in table 3 for this scenario.

With each algorithm 100 replicates of $\hat{\mathbf{X}}$ and thus $\mathbf{E}_{pr}$ were obtained for each of several tuning parameter(s): (λ, q) for l_qPG, and λ for HARD-IMPUTE and SOFT-IMPUTE.

Tables 1 and 2 show the minimum average $\mathbf{E}_{pr}$ with respect to the tuning parameter(s) for different algorithms. The l_qPG gives a better matrix estimator in terms of prediction accuracy (and $\mathbf{MSE}$ [15]) with the optimal $q \in (0, 1)$. From Table 3, l_qPG is also superior in terms of correct matrix rank recovery. The contour plot examples in Fig.1, 2 and 3 were obtained by l_qPG and show the average $\mathbf{E}_{pr}$ and rank respectively for each (λ, q). The optimal (λ, q) in Fig.1 and 2 give the minimum average $\mathbf{E}_{pr}$ shown in tables 1 and 2.

5. CONCLUSION

We have developed a simple algorithm l_qPG based on the MM technique for solving the l_q matrix completion problem $q \in (0, 1)$. In the simulation examples it was shown that the l_q rank penalty with $q \in (0, 1)$ was very competitive and provided improvement over the nuclear norm and rank restricted penalties in terms of matrix rank recovery and model prediction accuracy.

6. APPENDIX

Proof: (a) The proof is not as straight forward because it involves careful consideration of cases for σ_i to find the *correct* thresholding parameter.

Let ϕ' denote the derivative of ϕ, $d_i > 0$. Also, let $\alpha \triangleq [\mu q(1-q)]^{\frac{1}{2-q}}$ (> 0 and $< \delta$), $\beta \triangleq \alpha + \mu q \alpha^{q-1}$ and $d_{opt}(\sigma_i)$ be the global minimizer of ϕ, $d_i \geq 0$.

Now, ϕ' has a unique global minimum at α and is strictly increasing in d_i on both sides of α: details are in [15]. Since the stationary points of ϕ are given by the roots of ϕ', there are two cases to consider: when $\phi'(\alpha) \geq 0$ and $\phi'(\alpha) < 0$, i.e. when $\min \phi'$ is above and below the d_i-axis respectively.

Considering $\phi'(\alpha) \geq 0$: this means that either ϕ' is positive $\forall\ d_i > 0$ or that α is a stationary point of inflection of ϕ. In either case, the global minimizer of ϕ is at 0 (the boundary). By re-arranging, $\phi'(\alpha) \geq 0$ is equivalent to $\sigma_i \leq \beta$. Thus $d_{opt}(\sigma_i) = 0$ for $\sigma_i \leq \beta$.

Considering $\phi'(\alpha) < 0$: since $\min \phi'$ is below the d_i-axis, ϕ' has a unique root on (α, ∞), i.e. $\bar{x}(\sigma_i) \in (\alpha, \sigma_i)$, which is the only local minimizer of ϕ. Another root on $(0, \alpha)$ can only be a local maximizer of ϕ: details are in [15]. Therefore, it turns out that the global minimum of ϕ is either at $\bar{x}(\sigma_i)$ or at 0 (the boundary) depending on which gives a lower ϕ value. Now, $\phi(\bar{x}(\sigma_i)) < \phi(0)$ is equivalent to $\bar{x}(\sigma_i) > \delta$ (δ is from (4)) and by re-arranging, $\phi'(\alpha) < 0$ is equivalent to $\sigma_i > \beta$. Thus:

$$d_{opt}(\sigma_i) = \bar{x}(\sigma_i)\mathbf{I}(\bar{x}(\sigma_i) > \delta),\ \sigma_i > \beta \tag{8}$$

noting the discontinuity when $\bar{x}(\sigma_i) = \delta$, which by re-arranging is equivalent to $\sigma_i = h$ (h is from (4)). Since $h > \beta$, and $\hat{x}(\sigma_i)$ is strictly increasing in σ_i for $\sigma_i > \beta$: details are in [15], using (8) we have $d_{opt}(\sigma_i) = 0$ for $\beta < \sigma_i \leq h$ and $d_{opt}(\sigma_i) = \bar{x}(\sigma_i) \in (\delta, \sigma_i)$ for $\sigma_i > h$. Therefore, $d_{opt}(\sigma_i) = \tau(\sigma_i)$ in (5).

(b) The map ρ maps elements from $[\delta, \sigma_i]$ to $[\delta, \sigma_i]$ and is a contraction mapping on $[\delta, \sigma_i]$ complete. The result then follows by the application of the standard Banach Fixed Point Theorem: details are in [15].

7. REFERENCES

[1] R.H.Keshavan, S.Oh, and A.Montanari, "Matrix completion from a few entries," *CoRR, abs*/0901.3150, 2009.

[2] J.Rennie and N.Srebro, "Fast maximum margin matrix factorization for collaborative prediction," *22nd ICML*, pp. 713–719, 2005.

[3] N.Linial, E.London, and Y.Rabinovich, "The geometry of graphs and some of its algorithmic applications," *Combinatorica*, vol. 15, pp. 577–591, 1995.

[4] G.Obozinski, B.Taskar, and M.Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems," *Stat. Comput.*, vol. 20, pp. 231–252, 2010.

[5] R.Mazumder, T.Hastie, and R.Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *J. Mach. Learn. Res.*, vol. 11, pp. 2287–2322, 2010.

[6] C.Beck and R.DAndrea, "Computational study and comparisons of lft reducibility methods," *ACC*, pp. 1013–1017, 1998.

[7] M.Fazel, "Matrix rank minimization with applications," 2002, PhD thesis, Stanford University.

[8] E.J.Cand'es and T.Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE T. Inform. Theory*, vol. 56, pp. 2053–2080, 2009.

[9] D.Goldfarb and S.Ma Z.Wen, "Solving low-rank matrix completion problems efficiently," *47th Allerton conf. commun. cont. comp.*, pp. 1013–1020, 2009.

[10] R.Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc. B*, vol. 58, pp. 267–288, 1996.

[11] R.Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Proc. Let.*, vol. 14, pp. 707–710, 2007.

[12] J.Fan and R.Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. Am. Stat. Assoc.*, vol. 96, pp. 1348–1360, 2001.

[13] R.Saab and O.Yilmaz, "Sparse recovery by non-convex optimization-instance optimality," *Appl. Comput. Harmon. A.*, vol. 29, pp. 30–48, 2010.

[14] A.Beck and M.Teboulle, "A fast iterative shrinkage thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, pp. 183–202, 2009.

[15] G.Marjanovic and V.Solo, "On l_q optimization and matrix completion," *UNSW Technical Report*, August 2011, will be submitted to the IEEE T. Signal Proces.

[16] A.Majumdar and R.K.Ward, "Some empirical advances in matrix completion," *Signal Process.*, vol. 91, pp. 1334–1338, May 2011.