Statistical Science
2018, Vol. 33, No. 2, 238–260
https://doi.org/10.1214/18-STS642

Flexible Low-Rank Statistical Modeling with Missing Data and Side Information

William Fithian and Rahul Mazumder

Abstract. We explore a general statistical framework for low-rank modeling of matrix-valued data, based on convex optimization with a generalized nuclear norm penalty. We study several related problems: the usual low-rank matrix completion problem with flexible loss functions arising from generalized linear models; reduced-rank regression and multi-task learning; and generalizations of both problems where side information about rows and columns is available, in the form of features or smoothing kernels. We show that our approach encompasses maximum a posteriori estimation arising from Bayesian hierarchical modeling with latent factors, and discuss ramifications of the missing-data mechanism in the context of matrix completion. While the above problems can be naturally posed as rank-constrained optimization problems, which are nonconvex and computationally difficult, we show how to relax them via generalized nuclear norm regularization to obtain convex optimization problems. We discuss algorithms drawing inspiration from modern convex optimization methods to address these large scale convex optimization computational tasks. Finally, we illustrate our flexible approach in problems arising in functional data reconstruction and ecological species distribution modeling.

Key words and phrases: Matrix completion, nuclear norm regularization, matrix factorization, convex optimization, missing data.

1. INTRODUCTION

Matrix completion (Candès and Recht, 2009, Mazumder, Hastie and Tibshirani, 2010, Bennett and Lanning, 2007) via low-rank matrix modeling is a central problem in modern multivariate statistics and machine learning. This is due in large part to the advent of collaborative filtering and recommender systems (Agarwal and Chen, 2015), but matrix completion and matrix factorization applications extend to fields as diverse as image processing (Angst, Zach and Pollefeys, 2011), X-ray crystallography (Candès et al., 2015), seismology (Yang, Ma and Osher, 2013), and political science (Martin and Quinn, 2002, Gerrish and Blei, 2011).

In matrix completion, we partially observe a response matrix $Y \in \mathbb{R}^{n\times m}$, where each entry Y_{ij} represents a binary, categorical, or real-valued outcome. Typically each row and each column represents an entity of interest such as a user, a test question, an advertisement, or a point in time, and the response Y_{ij} is a result of some interaction between the ith row entity and the jth column entity. Typically, only a sparse subset of entries $\Omega \subseteq \{1,\dots,n\} \times \{1,\dots,m\}$ are observed, and the analyst's goal is to predict the missing entries as accurately as possible. To make progress, a common modeling assumption is that each row and column entity can be represented in a latent space of dimension $r \ll n, m$. If $u_i, v_j \in \mathbb{R}^r$ are the latent representations of row i and column j respectively, then any row–column interaction between i and j is assumed to

William Fithian is Assistant Professor, Department of Statistics, University of California, Berkeley, 301 Evans Hall, Berkeley, California 94720, USA (e-mail: wfithian@berkeley.edu). Rahul Mazumder is Assistant Professor, Sloan School of Management, Operations Research Center and MIT Center for Statistics, Massachusetts Institute of Technology, Building E62-583, 100 Main Street, Cambridge, Massachusetts 02142, USA (e-mail: rahulmaz@mit.edu).

depend only on the inner product $u_i'v_j$. Matrix completion has attracted a great deal of attention in the statistical machine learning community in recent years; see, for example, Candès and Recht (2009), Candès and Tao (2010), Candes and Plan (2010), Keshavan, Montanari and Oh (2010), Mazumder, Hastie and Tibshirani (2010). A parallel line of work in the multivariate statistics literature discusses iterative methods to deal with missing values encountered in classical tasks such as principal components analysis (Josse and Husson, 2012), correspondence analysis (de Leeuw and van der Heijden, 1988) multiple correspondence analysis, and multivariate analysis of mixed data sets (Audigier, Husson and Josse, 2016).

Matrix completion became a major focal point of methodological and applied machine learning research in part due to the famous Netflix Prize competition of 2006–2009 (Bennett and Lanning, 2007, Bell and Koren, 2007), in which researchers competed to improve on Netflix's baseline algorithm for recommending movies to users. In the Netflix data, row i is a particular Netflix user and column j is a movie, and the observed entries are movie ratings from 1–5. Y_{ij} is observed if user i has assigned a rating to movie j; otherwise we can interpret it as the rating that user i *would assign* to movie j if she were to watch and rate it. Among approximately 480K users and 20K movies, on average each user rates only 200 movies leading to $|\Omega| \approx 10^8$ observed entries out of $nm \approx 10^{10}$ total entries. An objective in such movie-recommender systems is to recommend movies that their users would enjoy.

As we discuss in Section 2.1, many of the most familiar methods in classical multivariate statistics including principal components analysis, reduced-rank regression, multidimensional scaling, canonical correlation analysis and correspondence analysis can be viewed as estimating lower-dimensional latent factors u_i and v_j to approximate a fully observed matrix under a generalized least-squares loss criterion. These classical methods can flexibly incorporate diverse types of side information about row and column entities by imposing constraints or quadratic penalties on the latent factors, with the (truncated) singular value decomposition (SVD) (Golub and Van Loan, 1983) providing a generic tool for optimizing quadratic objectives with rank constraints. However, if either (a) the matrix is not fully observed or (b) the objective is not quadratic, this optimization framework breaks down. This is because the associated nonconvex optimization problems become difficult due to the presence of the rank constraint; and solutions to these problems can no longer be obtained via a simple SVD. Global optimization of nonconvex problems involving rank-constraints are problematic from an algorithmic viewpoint (see, for example, Bertsimas, Copenhaver and Mazumder, 2017 and references therein). Inspired mainly by the success of nuclear norm regularization methods in matrix completion, we propose to explore convex relaxations of the rank constraint and study the associated convex optimization problems as surrogates to the nonconvex rank constrained problems. These lead to instances of semidefinite optimization problems (Boyd and Vandenberghe, 2004) which are challenging to scale to large scale instances.

Our main aim in this article is to explore an alternative computational and modeling framework, based on convex optimization with a *generalized nuclear norm* penalty. This viewpoint expands the scope of the SVD framework described above, adapting it to the more general setting where some entries are missing or the loss function is not quadratic. Section 1.2 reviews low-rank approximation and the relationship between the singular value decomposition, the nuclear norm and quadratic regularization in the row and column latent factors, and shows how we can flexibly impose modeling assumptions on the latent variables in matrix completion problems. Section 4 reviews algorithmic options for these large scale semidefinite optimization problems.

Despite their advantages, convex (semidefinite) optimization approaches are often dismissed in the collaborative filtering literature as impractical for large problems since their worst-case scaling is poor and vanilla implementations are not practical; see, for example, Salakhutdinov and Mnih (2008b), Menon and Elkan (2010). However, despite the poor worst-case performance, there are rich classes of models in which a generalized version of nuclear norm regularization scales well with problem size, as we will see in Section 4. Finally, to predict missing data as well as possible, we must understand the *missing data mechanism*; that is, we must understand why the data are missing. While the missing-data mechanism has received comparatively little attention in the machine learning literature on matrix completion, the winning team in the Netflix prize reported they saw a breakthrough in their prediction error once they appreciated that users are not selecting movies wholly at random, and the choice of which movies to watch may be quite informative about the user's latent type (Bell and Koren, 2007). By

contrast, there is a greater awareness of the missing-data mechanism in the multivariate statistics literature; see Audigier, Husson and Josse (2016) for a discussion. Section 3 discusses how assumptions about the missing-data mechanism can be incorporated into our low-rank modeling framework and how they may complicate the interpretation of our predictions.

Organization of paper: The remainder of the paper is organized as follows. Section 2 presents a unified framework of modeling with low-rank and nuclear norm regularization for several problems in classical multivariate statistics and modern collaborative filtering applications—we also draw parallels with Bayesian modeling schemes. Section 3 presents a connection between the classical missing data literature in statistics and the matrix completion framework. Section 4 presents a broad overview of optimization algorithms that can be used for the class of problems studied herein. Section 5 presents some illustrative examples.

1.1 Preliminaries and Notation

For a matrix A, we will generically denote the ith row with a lower-case letter a_i'. We denote 1_n as the length-n vector with 1 in every coordinate, and for a matrix A we write the *Moore–Penrose pseudo-inverse* as A^+. Multiplying AA^+ we obtain the projection matrix into the column space of A, which we will denote as Π_A, and we denote the projection into its orthogonal complement as $\Pi_A^\perp = (\mathbb{I} - \Pi_A)$, where $\mathbb{I}$ is the identity matrix. We assume without loss of generality that $n \geq m$. We write the singular values of a matrix A as $\sigma_1(A), \ldots, \sigma_m(A)$, arranged in decreasing order. Let $\|A\|_F^2 = \mathrm{Tr}(A'A) = \sum_{ij} A_{ij}^2 = \sum_i \|a_i\|_2^2$ denote the *Frobenius norm* of matrix A. Let $\|A\|_* = \sum_{k=1}^m \sigma_k(A)$ denote the *nuclear norm* of A, or the sum of its singular values. By contrast $\|A\|_F^2$ is equal to the sum of its squared singular values, and $\mathrm{rank}(A)$ is the number of nonzero singular values. In this sense, the rank, nuclear norm, and Frobenius norm are natural matrix (spectral) counterparts of the ℓ_0, ℓ_1, and ℓ_2 norms for vectors.

1.2 Low-Rank Approximation and the Nuclear Norm

Our mathematical point of departure is the prototypical problem of *low-rank approximation* of a matrix Y in $\mathbb{R}^{n\times m}$, wherein we attempt to find r-dimensional row and column variables $u_i, v_j \in \mathbb{R}^r$ such that $Y_{ij} \approx u_i'v_j$. In matrix notation, $Y \approx UV'$ for $U \in \mathbb{R}^{n\times r}$, $V \in \mathbb{R}^{m\times r}$, where the quality of approximation is generically measured in terms of some loss function $\mathcal{L}(\cdot; Y)$:

$$\underset{U\in\mathbb{R}^{n\times r}, V\in\mathbb{R}^{m\times r}}{\text{minimize}} \ \mathcal{L}(UV'; Y). \tag{1}$$

Note that in (1), the row and column variables u_i, v_j are intrinsically unidentifiable: for any invertible matrix $A \in \mathbb{R}^{r\times r}$ we could replace U with $\widetilde{U} = UA'$ and V with $\widetilde{V} = VA^{-1}$. Then $\widetilde{U}\widetilde{V}' = UV'$, leading to the same loss. To eliminate this ambiguity, we can introduce the optimization variable $\Theta = UV'$, constraining its rank to be at most r, leading to

$$\underset{\Theta\in\mathbb{R}^{n\times m}}{\text{minimize}} \ \mathcal{L}(\Theta; Y) \quad \text{s.t.} \quad \mathrm{rank}(\Theta) \leq r. \tag{2}$$

Certain constraints or penalties that we might impose on U or V translate to constraints or penalties on Θ. For example, in reduced-rank regression we have a feature matrix $X \in \mathbb{R}^{n\times d}$ and require $U = XB$ for some $B \in \mathbb{R}^{d\times r}$; in terms of Θ, it is equivalent to require that $\Pi_X^\perp \Theta = 0$. Once we find the best fitting Θ, decomposing it into row variables U and column variables V is essentially a matter of interpretation; if prediction is our aim, it is enough to estimate Θ.

If $\mathcal{L}(\Theta; Y)$ is the simple least squares loss $\|Y - \Theta\|_F^2$, it is well known that (2) is solved by the rank-r truncated SVD (Golub and Van Loan, 1983) of Y. That is, $\Theta = U^r D^r V^{r\prime}$, where the columns of $U^r \in \mathbb{R}^{n\times r}$ and $V^r \in \mathbb{R}^{m\times r}$ consist, respectively, of the first r left and right singular vectors of Y, and $D^r \in \mathbb{R}^{r\times r}$ is diagonal with $D_{kk}^r = \sigma_k(Y)$. If the eigenvalues of Y are all distinct, then the decomposition is unique for every r, up to reversing signs of the columns of U^r and V^r. As we will see in Section 2.1, the truncated SVD provides a powerful computational framework for incorporating flexible side information and modeling assumptions about the row and column variables.

Unfortunately, if $\mathcal{L}(\cdot;\cdot)$ departs from the Frobenius norm, then Problem (2), a nonconvex problem owing to the nonconvexity of the rank constraint, is generally computationally intractable. In matrix completion, we can only measure errors on the observed entries; thus the natural least-squares loss is $\mathcal{L}(\Theta; Y) = \sum_{(i,j)\in\Omega}(Y_{ij} - \Theta_{ij})^2$. This seemingly minor variant of the Frobenius norm makes Problem (2) difficult to solve using a simple SVD. In addition to matrix completion, non-Frobenius loss functions are strongly motivated in several other settings including:

1. *Sparsely observed functional data*: We are given noisy observations $Y_{ij} = g_i(t_{ij}) + \epsilon_{ij}$ for a smooth function g_i at various locations in a temporal, spatial or other domain, and we wish to reconstruct the functions g_i at unobserved locations. t_j could be values on

a real interval or can correspond to spatial locations. This setting is similar to matrix completion, but with notionally infinitely many "column entities" representing locations in the domain that may not occur at all in the entire data set; as a result smoothness assumptions are crucial to recover identifiability. We revisit this setting in Section 5.1.

2. *Exponential families*: If Y_{ij} arises from an exponential family model such as binomial, Poisson, etc., we can model its natural parameter Θ_{ij} as arising from a low-rank model (Roweis, 1998, Rennie and Srebro, 2005, Srebro, Rennie and Jaakkola, 2005). Further generalizing this approach, Yee and Hastie (2003) suggest *reduced-rank vector generalized linear models* (RR-VGLMs) in which $\Theta = XB$ for some observed covariates $X \in \mathbb{R}^{n\times d}$.

3. *Robust loss functions*: If some of the Y_{ij} values are outliers (or arise from a heavy tailed distribution), we may choose to replace the squared error loss with a more robust entry-wise loss function such as the *least absolute deviation loss* $\mathcal{L}(\Theta; Y) = \sum_{ij} |\Theta_{ij} - Y_{ij}|$ or the *Huber loss* (Huber, 1996) $\mathcal{L}(\Theta; Y) = \sum_{ij} \rho_\gamma(\Theta_{ij} - Y_{ij})$ where

$$\rho_\gamma(x) = \begin{cases} \frac{1}{2}x^2, & |x| \le \gamma, \\ \gamma\left(|x| - \frac{1}{2}\gamma\right), & |x| > \gamma \end{cases}$$

is a quadratic function for small x but linear in the tails. The least absolute deviation loss (formulated differently), paired with the nuclear norm relaxation (discussed below), forms the basis for the celebrated *robust PCA* method of Candès et al. (2011).

Although there are many specialized algorithms for finding approximate solutions or local minima to such models (given by nonconvex optimization problems), there is no guarantee that we can obtain a global minimum of the problem as posed. A well-designed method may find a suitable local solution or saddle point for many problems, but it can be difficult to predict how these specialized algorithms will perform once we modify the problem to incorporate side information.

1.2.1 *Nuclear norm regularization.* By analogy to the Lasso (Tibshirani, 1996) relaxation of sparse regression, Fazel (2002) propose a convex relaxation scheme replacing the rank constraint with a constraint on the nuclear norm, leading to the convex semidefinite optimization problem:

$$\min_{\Theta} \mathcal{L}(\Theta; Y) \quad \text{s.t.} \quad \|\Theta\|_* \le \delta, \tag{3}$$

or in Lagrangian form,

$$\min_{\Theta} \mathcal{L}(\Theta; Y) + \lambda\|\Theta\|_*. \tag{4}$$

Like the Lasso in linear regression, the nuclear norm plays two roles: first, it promotes a low-rank solution by setting many of the singular values of Θ to zero; and second, it regularizes the low-rank solution by shrinking the singular values of Θ toward zero. If $\mathcal{L}(\Theta; Y) = \frac{1}{2}\|Y - \Theta\|_F^2$ then the problem is solved by the *nuclear-norm soft thresholding operator* or the *singular value thresholding operator*

$$S_\lambda(Y) \in \operatorname*{argmin}_{\Theta} \frac{1}{2}\|Y - \Theta\|_F^2 + \lambda\|\Theta\|_*, \tag{5}$$

where $S_\lambda(A)$ is defined as $\operatorname{diag}((A_{11} - \lambda)_+, \ldots, (A_{rr} - \lambda)_+)$ if A is a diagonal $r \times r$ matrix, and otherwise $S_\lambda(A) = US_\lambda(D)V'$ where UDV' is the (full-rank) SVD of A, with D a diagonal matrix. Soft-thresholding the singular values leads to low-rank solutions, for large values of λ. That is, rather than directly constraining the rank, we add a nuclear norm penalty that favors low-rank solutions.

The nuclear norm may alternatively be viewed as a regularization applied to the latent factors U and V, which can be seen from the following identity appearing in Fazel (2002), Srebro, Rennie and Jaakkola (2005):

$$\|\Theta\|_* = \min_{U,V:UV'=\Theta} \left\{\frac{1}{2}\|U\|_F^2 + \frac{1}{2}\|V\|_F^2\right\}. \tag{6}$$

Let $\hat{\Theta}$ be an optimal solution to Problem (4) with $\hat{r} = \operatorname{rank}(\hat{\Theta})$. In the light of (6) and observing $\hat{\Theta}$ has low rank, it is easy to see that the following optimization problem:

$$\min_{U\in\mathbb{R}^{n\times r}, V\in\mathbb{R}^{m\times r}} \mathcal{L}(UV'; Y) + \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2 \tag{7}$$

is equivalent to Problem (4) for any $r \ge \hat{r}$; and $\hat{\Theta} = \hat{U}\hat{V}'$ where $(\hat{U}, \hat{V})$ is a minimizer of Problem (7)—see Hastie et al. (2015), and references therein. We note that Problem (7) for varying r and λ, leads to a richer class of problems[1] than the convex formulation (4). Depending upon the context or modeler's preference, it may be reasonable to consider Problem (7) for $r < \hat{r}$. However, in the latter case, it may not be possible to obtain a global minimizer to Problem (7) in a

[1]This will happen as soon as we take r in Problem (7) to be smaller than $\hat{r}$.

tractable fashion. Criterion (7) lends itself to nice interpretation: If $\mathcal{L}(\cdot; Y)$ is a negative log-likelihood function or can be interpreted as such (Tipping and Bishop, 1999), then Problem (7) is a maximum a posteriori (MAP) estimation criterion with independent Gaussian priors on the entries of U and V (Salakhutdinov and Mnih, 2008b, Angst, Zach andPollefeys, 2011, Menon and Elkan, 2010). Problem (7) is an optimization problem in (U, V)—and if r is small—performing simple gradient descent or block coordinate descent methods on Problem (7) is quite simple. A caveat however, is that Problem (7) is *not* convex in (U, V) (owing to the product terms involving entries in U and V)—thus one may get stuck in poor stationary points/saddle points. However, as we discuss in Section 4.1.2; under some conditions, a locally optimal solution to Problem (7) corresponds to the global minimum of Problem (4) (assuming of course that $r \geq \hat{r}$). Note that $\hat{r}$ or a tight estimate of it is not known before solving Problem (4)—however, it can be estimated via iterative schemes as a part of the optimization algorithm (see Section 4.1.2 for details).

Interestingly, solutions to Problem (3) often approximate solutions to Problem (2) quite well. In their seminal work, Candès and Recht (2009) and Candès and Tao (2010) study the noiseless matrix completion problem showing that the nuclear norm leads to exact recovery of an underlying low-rank matrix under coherence like assumptions on the underlying matrix even when a few entries of the matrix are observed. Candes and Plan (2010) and Negahban and Wainwright (2012) study theoretical properties of the noisy matrix completion problem using nuclear norm regularization.

Theoretical properties of Problems (3) and (4) for loss functions beyond squared error have been studied by several authors. For example, Davenport et al. (2014) study the problem of *one-bit matrix completion*, where the response is binary and the entry wise loss is logistic, with an additional ℓ_∞-norm constraint on the entries of the matrix; and Lafond (2015) study prediction error bounds for matrix completion for exponential family models with a nuclear norm penalty. Carpentier et al. (2016) discuss confidence sets for the low-rank matrix completion problem and Klopp et al. (2015) consider a multinomial matrix completion problem where the observed entries are quantized with a few levels (in their framework the missingness need not be uniform). They study a regularized negative log-likelihood problem, where the latent variables are regularized by a nuclear norm penalty and an additional constraint on the maximal absolute entries of the matrix. Udell et al. (2016) extend a previous version of this manuscript (Fithian and Mazumder, 2013) and discuss computational aspects of low-rank modeling arising in machine learning problems.

1.2.2 *The generalized nuclear norm.* We can generalize the penalty in Problem (4) to consider the following optimization problem:

$$\min_{\Theta} \mathcal{L}(\Theta; Y) + \lambda \| P\Theta Q \|_* \tag{8}$$

for a priori specified (possibly data dependent) positive semidefinite matrices P and Q. We can interpret P and Q in Problem (8) as modulating the degree of ℓ_2 penalization for U and V respectively, by penalizing some directions more than others. For example, if $X \in \mathbb{R}^{n\times d}$ is a matrix of features for the rows then we might use $P = \mathbb{I} - \Pi_X$ (where $\mathbb{I}$ is the identity matrix) so that the component of Θ explained by X is unpenalized (see Section 2.2).

Several other authors have proposed interesting specific applications of the generalized nuclear norm—Salakhutdinov and Srebro (2010) advocate a special case of Problem (8) with diagonal P and Q, and Angst, Zach andPollefeys (2011) apply the generalized nuclear norm to the structure-from-motion problem in computer vision. Abernethy et al. (2009) frame collaborative filtering in very general terms of estimating compact linear operators in Hilbert space. Their proposals for regularizing Θ have the most overlap with ours but with less focus on scalable computation.

Provided that $\Theta \mapsto \mathcal{L}(\Theta; Y)$ is convex, Problems (3), (4), and (8) can be solved in polynomial time using standard convex optimization techniques for semidefinite optimization problems (Boyd and Vandenberghe, 2004). Convex optimization is appealing because it allows abstraction of our statistical model from our estimation algorithm. Even so, the computational cost of off-the-shelf interior point solvers become prohibitively large as soon as the problem sizes become larger than a few hundred. Toward this end, first order[2]

[2]First-order optimization algorithms are iterative methods with significantly low per iteration cost when compared to Interior Point algorithms. Even if first order methods take many more iterations than an Interior Point algorithm to converge to a solution with comparable accuracy, their low-memory requirement and cheap per iteration cost makes them applicable to modern large scale problem instances. In addition, low to moderate accuracy solutions lead to excellent estimates with good statistical properties especially in large noisy datasets (Bottou and Bousquet, 2008).

methods (Nesterov, 2004) are used to obtain low-to-moderate accuracy solutions. Indeed, as we discuss in Section 4, developing fast, scalable and rigorous algorithms for nuclear norm regularized problems continues to be an active area of research, across the fields of statistics, machine learning and optimization.

If P and Q are invertible then a simple change of variables argument shows that we can rewrite Problem (8) as $\mathcal{L}(UV'; Y) + \frac{\lambda}{2}\|PU\|_F^2 + \frac{\lambda}{2}\|QV\|_F^2$. In fact, the same holds for generic semidefinite P and Q, as we state below in Proposition 1 (the proof is in Appendix A).

PROPOSITION 1. *Let $P \in \mathbb{R}^{n\times n}$ and $Q \in \mathbb{R}^{m\times m}$ be positive semidefinite. Then for any function $\mathcal{L}(\cdot; Y) : \mathbb{R}^{n\times m} \to \mathbb{R}$, we have*

$$\inf_{U\in\mathbb{R}^{n\times r}, V\in\mathbb{R}^{m\times r}} \mathcal{L}(UV'; Y) + \frac{1}{2}\|PU\|_F^2 + \frac{1}{2}\|QV\|_F^2 \tag{9}$$

$$= \inf_{\Theta} \mathcal{L}(\Theta; Y) + \|P\Theta Q\|_* \tag{10}$$

$$= \inf_{\Theta_1,\Theta_2,\Theta_3} \mathcal{L}\big(P^+\Theta_1 Q^+ + \Pi_P^\perp \Theta_2 + \Theta_3 \Pi_Q^\perp; Y\big) + \|\Theta_1\|_* \tag{11}$$

for any $r \geq \operatorname{rank}(\hat{\Theta})$ with $\hat{\Theta}$ being a minimizer of (10); *and P^+ and Q^+ are the Moore–Penrose pseudo-inverses of P and Q, and $\Pi_P^\perp$ and $\Pi_Q^\perp$ are projections onto their respective null spaces.*

Proposition 1 is useful because it allows us to move easily back and forth between modeling latent factors via the more interpretable formulations (9)–(10) and Problem (11). As we will discuss further in Section 4, when $\Theta \mapsto \mathcal{L}(\Theta; Y)$ is convex and smooth, formulation (11) is often computationally attractive for two reasons. First, we may be able to represent Θ_2 and Θ_3 as matrices of much smaller dimension, while Θ_1 is low-rank and can be represented efficiently as an outer product of smaller matrices. If $P = \Pi_X^\perp$, for example, then $\Pi_P^\perp = \Pi_X$, and we can replace the unpenalized term $\Pi_P^\perp\Theta_2$ with XB, where $B \in \mathbb{R}^{d\times m}$. Second, if we use proximal gradient descent then the proximal steps for Θ_1 can be solved using a soft-thresholded SVD (by contrast, proximal gradient steps with respect to a generalized nuclear norm generically cannot be solved in closed form). We refer the reader to Section 4 for further details pertaining to the computational aspects of this problem; and its special cases.

1.3 Other Approaches

The max-norm regularization: The max-norm is another convex proxy for the rank of a matrix that is often used in the context of matrix completion and related problems (Srebro, Rennie and Jaakkola, 2005, Srebro and Shraibman, 2005). Convex and closely related to the nuclear norm, the max-norm of a matrix Θ can be defined via matrix factorizations as:

$$\|\Theta\|_{\max} = \min_{U,V:\Theta=UV'} \big(\|U\|_{2,\infty}\|V\|_{2,\infty}\big), \tag{12}$$

where $\|A\|_{2,\infty}$ denotes the maximum ℓ_2 row norm of the matrix A, that is, $\|A\|_{2,\infty} = \max_i(\sum_j a_{ij}^2)^{1/2}$. Lee et al. (2010) demonstrate that the empirical performance of a max-norm regularized version of matrix completion may lead to better predictive performance on some collaborative filtering datasets. Theoretical properties of the max-norm have been studied by Srebro, Rennie and Jaakkola (2005), Srebro and Shraibman (2005), Foygel and Srebro (2011), Cai and Zhou (2013), but it seems that max-norm regularization is computationally more challenging and much less studied than the nuclear-norm counterpart.

Bayesian methods: Another way to incorporate domain knowledge or side information is to use complex hierarchical Bayes models which can be fit using various specialized approaches. For example, Salakhutdinov and Mnih (2008a) study a generative model with additional priors on the hyper-parameters and develop a Gibbs sampling scheme for the problem, leading to a computationally intensive method requiring 200 hours to train a model with $r = 60$ on the Netflix dataset. Aggarwal and Chen (2009) also propose a more general Bayesian modeling framework which we revisit later in Section 2.2. Agarwal, Zhang and Mazumder (2011) study an example where these covariance matrices are unknown and they are estimated via inverse covariance matrix estimation. Todeschini, Caron and Chavent (2013) place a prior on the singular values of the matrix and propose an EM-stylized algorithm for the task. Cottet and Alquier (2018) study the one-bit matrix completion from a Bayesian perspective using variational techniques.

Because the resulting model specifications are highly nonconvex, doing tractable inference or making formal (computational) statements about the quality of the estimates obtained are rather challenging. However, as we show in Section 2.2, our generalized nuclear norm regularization framework can be used to perform MAP inference in these models. Even if our goal is to sample from the posterior rather than find the

MAP estimate, it is often practically useful to use optimization techniques to understand properties of the posterior distribution or help with the sampling. In this vein, Aggarwal and Chen (2009) empirically compare several methods for fitting the same model, and settle on an estimation method of their own devising, called Monte Carlo-EM, to find a local maximum of the marginal likelihood.

2. MODELING IN LATENT SPACE

2.1 Low-Rank Modeling with the SVD

Some of the most familiar methods in classical statistics amount to low-rank least-squares approximation of an appropriate matrix, along with some preprocessing of the matrix and postprocessing of the singular vectors. We review some examples here, for more details and a classical perspective see Mardia, Kent and Bibby (1979). A main purpose of this section is to reformulate well-known low-rank modeling tasks arising in the classical multivariate statistics literature into stylized optimization problems, to set the stage for the more elaborate modeling techniques described in Section 2.2.

Principal components analysis: Principal components analysis (PCA) computes the directions of greatest variation among rows of a data matrix. We begin by subtracting the mean from each column, obtaining the column-centered matrix $\widetilde{Y} = Y - n^{-1}1_n 1_n' Y = \Pi_{1_n}^{\perp} Y$. The first r principal components and principal component loadings are, respectively, the columns of $U^r D^r$ and V^r where $U^r D^r V^{r\prime}$ is the rank-r truncated SVD of $\widetilde{Y}$. If $\overline{Y} = n^{-1}Y'1_n$, the vector of column means, then we can reconstruct a least-squares approximation to Y as $\widehat{Y} = 1_n \overline{Y}' + U^r D^r V^{r\prime}$.

Consider modeling $Y_{ij} \overset{\text{ind.}}{\sim} N(\Theta_{ij}, \sigma^2)$ where $\Theta_{ij} = \beta_j + u_i' v_j$, or in matrix form $\Theta = 1_n \beta' + UV'$ ($1_n \in \mathbb{R}^n$ is a vector of all ones) where $\beta \in \mathbb{R}^m$, $U \in \mathbb{R}^{n\times r}$, and $V \in \mathbb{R}^{m\times r}$. Because UV' can be any matrix with rank less than r, we can equivalently write $\Theta = 1_n\beta' + \Gamma$ where $\Gamma \in \mathbb{R}^{n\times m}$ with $\operatorname{rank}(\Gamma) \le r$. In this model, the maximum likelihood estimator for Θ solves

$$\begin{aligned} &\min_{\Theta} \|Y - \Theta\|_F^2 \\ &\text{s.t.} \quad \Theta = 1_n\beta' + \Gamma, \quad \operatorname{rank}(\Gamma) \le r. \end{aligned} \tag{13}$$

Including the saturated column effect vector β explicitly as an unpenalized term guarantees that the fitted Γ matrix is a function only of the column-centered matrix $\widetilde{Y}$.

Note that for any solution (β, Γ) to Problem (13) with $1_n'\Gamma \neq 0$ (i.e., some column of Γ has nonzero mean), the alternate solution $(\beta + n^{-1}\Gamma' 1_n, \Pi_{1_n}^{\perp}\Gamma)$ leads to exactly the same Θ value, and hence the same likelihood. Because $\operatorname{rank}(\Pi\Gamma) \le \operatorname{rank}(\Gamma)$ for any projection matrix Π, we have no reason to entertain solutions with $1_n'\Gamma \neq 0$. Thus, we can add the constraint $\Pi_{1_n}\Gamma = 0$ without changing the estimation problem. As a result, we have $\Pi_{1_n}^{\perp}\Theta = \Gamma$ and $\Pi_{1_n}\Theta = 1_n\beta'$.

Eliminating β and Γ from the problem, we can rewrite it in condensed form as

$$\min_{\Theta} \|Y - \Theta\|_F^2 \quad \text{s.t.} \quad \operatorname{rank}(\Pi_{1_n}^{\perp}\Theta) \le r. \tag{14}$$

In other words, the rank constraint only applies to the portion of the column space of Θ that is orthogonal to 1_n. In that sense, we can say 1_n is an unregularized column direction.

Having derived Problem (14), we can easily solve for the maximum likelihood estimator by noting that

$$\begin{aligned} \|Y - \Theta\|_F^2 &= \|\Pi_{1_n} Y - \Pi_{1_n}\Theta\|_F^2 + \|\Pi_{1_n}^{\perp} Y - \Pi_{1_n}^{\perp}\Theta\|_F^2 \\ &= \|1_n\overline{Y}' - \Pi_{1_n}\Theta\|_F^2 + \|\widetilde{Y} - \Pi_{1_n}^{\perp}\Theta\|_F^2. \end{aligned}$$

We can set the first term to zero by taking $\Pi_{1_n}\Theta = 1_n\overline{Y}'$ (leading to $\beta = \overline{Y}$) and minimizing $\|\widetilde{Y} - \Pi_{1_n}^{\perp}\Theta\|_F^2$ via the SVD (leading to $\Gamma = UDV'$).

Reduced rank regression: As a second example, suppose we have a response matrix $Y \in \mathbb{R}^{n\times m}$ and feature matrix $X \in \mathbb{R}^{n\times d}$, and consider regressing each column of Y on the predictors X, but sharing information across the m responses via a rank constraint. That is, suppose we again model $Y_{ij} \overset{\text{ind.}}{\sim} N(\Theta_{ij}, \sigma^2)$, but now modeling $\Theta_{ij} = \alpha_j + x_i'\beta_j$, for $j = 1, \ldots, m$, and with a constraint on the rank of $B = [\beta_1 \cdots \beta_m]$, leading to the popular reduced-rank regression model (Anderson, 1951, Reinsel and Velu, 1998). The maximum likelihood problem can then be written as

$$\begin{aligned} &\min_{\Theta} \|Y - \Theta\|_F^2 \\ &\text{s.t.} \quad \Theta = 1_n\alpha' + XB, \quad \operatorname{rank}(B) \le r. \end{aligned} \tag{15}$$

By a similar logic as before, we may assume without loss of generality that the columns of X have mean zero: if $\widetilde{X} = \Pi_{1_n}^{\perp} X$ and (α, B) solves Problem (15) with data $(\widetilde{X}, Y)$ then $(\alpha + n^{-1}(XB)'1_n, B)$ solves Problem (15) with data (X, Y). Furthermore, noting that

$$\begin{aligned} &\{XB : B \in \mathbb{R}^{d\times m}, \operatorname{rank}(B) \le r\} \\ &\quad = \{A \in \mathbb{R}^{n\times m} : \Pi_X^{\perp} A = 0, \operatorname{rank}(A) \le r\}, \end{aligned}$$

we can eliminate α and B and obtain

$$\min_{\Theta} \|Y-\Theta\|_F^2 \quad \text{s.t.} \quad \operatorname{rank}\bigl(\Pi_{1_n}^{\perp}\Theta\bigr) \le r, \quad \Pi_{[1_n,X]}^{\perp}\Theta = 0. \tag{16}$$

In Problem (16), we see a similar prioritization of column directions as in Problem (14), only now with three levels of prioritization: 1_n is unregularized, the column space of X is regularized via a rank constraint, and all other directions are completely killed.

As before, we can solve Problem (16) by decomposing the Frobenius norm into the three column spaces of interest:

$$\|Y-\Theta\|_F^2 = \|\Pi_{1_n}Y - \Pi_{1_n}\Theta\|_F^2 + \|\Pi_X Y - \Pi_X\Theta\|_F^2 + \bigl\|\Pi_{[1_n,X]}^{\perp}Y\bigr\|_F^2.$$

We can eliminate the first term by taking $\alpha = \overline{Y}$. To minimize the second term, we set $\Pi_X\Theta = XB = UDV'$, the rank-r truncated SVD of $\Pi_X Y$, and solving for B we obtain $B = X^{+}UDV'$. The third term depends only on Y and does not influence the solution.

Non-identity covariance, row effects and further generalizations: The basic formulations of PCA and reduced-rank regression above are natural if the m columns of Y are measured in the same units and errors are of a comparable scale. In other cases, it would be more natural to measure the approximation error relative to a different metric based on the modified log-likelihood. For example, suppose that $Y_{ij} \overset{\text{ind.}}{\sim} N(\Theta_{ij}, \Sigma)$ for some known or estimated error covariance matrix $\Sigma \succ 0$.

In that case, the maximum-likelihood problem for PCA becomes

$$\min_{\Theta} \bigl\|(Y-\Theta)\Sigma^{-1/2}\bigr\|_F^2 \quad \text{s.t.} \quad \operatorname{rank}\bigl(\Pi_{1_n}^{\perp}\Theta\bigr) \le r. \tag{17}$$

Making the same decomposition as before, we will set $\beta = \overline{Y}$ to eliminate the residual in the direction of 1_n, but to minimize the second term the solution for Γ will solve

$$\min_{\Gamma} \bigl\|\widetilde{Y}\Sigma^{-1/2} - \Gamma\Sigma^{-1/2}\bigr\|_F^2 \quad \text{s.t.} \quad \operatorname{rank}(\Gamma) \le r. \tag{18}$$

To solve Problem (18) we can simply change variables to $\widetilde{\Gamma} = \Gamma\Sigma^{-1/2}$, noting that $\operatorname{rank}(\Gamma) = \operatorname{rank}(\widetilde{\Gamma})$ for any $\Gamma \in \mathbb{R}^{n\times m}$. Then we see the minimizer is $\widetilde{\Gamma} = UDV'$, the rank-$r$ truncated SVD of $\widetilde{Y}\Sigma^{-1/2}$, and $\Gamma = UDV'\Sigma^{1/2}$. Using a further change of variables, we could also handle the more general case where the loss function is replaced by $\|\Phi^{-1/2}(Y-\Theta)\Sigma^{-1/2}\|_F^2$.

Generalizing in another direction, we might wish our model to incorporate row-wise fixed effects in addition to column-wise fixed effects, but with low-rank interactions between rows and columns. In that case, we might model $\Theta_{ij} = \alpha_i + \beta_j + \Gamma_{ij}$, leading to the likelihood criterion

$$\min_{\Theta} \|Y-\Theta\|_F^2 \quad \text{s.t.} \quad \Theta = \alpha 1_m' + 1_n\beta' + \Gamma, \quad \operatorname{rank}(\Gamma) \le r. \tag{19}$$

By a similar argument as in the previous section, we can assume without loss of generality that both $1_n'\Gamma = 0$ and $\Gamma 1_m = 0$, leading to the condensed criterion

$$\min_{\Theta} \|Y-\Theta\|_F^2 \quad \text{s.t.} \quad \operatorname{rank}\bigl(\Pi_{1_n}^{\perp}\Theta\Pi_{1_m}^{\perp}\bigr) \le r. \tag{20}$$

In this case, there is an unregularized column direction *and* an unregularized row direction; the rank constraint only applies to the portion of the model orthogonal to both. The answer can still be computed in closed form via a truncated SVD of $\Pi_{1_n}^{\perp}\Theta\Pi_{1_m}^{\perp}$.

By combining and extending the ideas above, many further generalizations are possible. As long as we use a generalized least-squares loss function of the form $\|\Phi^{-1/2}(Y-\Theta)\Sigma^{-1/2}\|_F^2$, we can choose from a great variety of models for Θ that are all computable in closed form using a common computational framework based on the SVD. Unfortunately, a simple SVD is no longer sufficient to get solutions for more general loss functions. For matrix completion with squared error loss, for example, a closed form solution cannot be obtained via a low-rank SVD. The computational difficulty in these problems stems from the presence of the rank constraint resulting in notoriously difficult optimization problems. One way to bypass this computational difficulty is to *relax* the rank constraint into the convex nuclear norm constraint—thereby leading to convex optimization problems, which can (in theory) be solved in polynomial time.

2.2 Low-Rank Modeling with Nuclear-Norm Regularization

When we move from the rank-constrained problem to the nuclear-norm-regularized problem we can use essentially all of the same manipulations as in the previous section to reduce any constraints on U and V to constraints on Θ. In addition to constraining the latent factors, however, we have an additional option to impose Bayesian priors on the factors and fit the resulting models by MAP estimation. We again discuss several examples below. The basic statistical models explored in this section have appeared before, especially in the context of Bayesian hierarchical models (Aggarwal and Chen, 2009) (see also references therein) popularly

used in the data-mining community in recommender systems. However, we provide a new perspective on the MAP estimation of these problems—exploring how they can be tied to convex optimization problems involving the generalized nuclear norm.

2.2.1 *Matrix completion*. In matrix completion problems like the Netflix challenge, a simple and appealing model is to assign a marginal effect (Koren, Bell and Volinsky, 2009, Hastie et al., 2015) to each row and column entity (e.g., a movie's overall quality and a user's overall affect) as well as a low-rank interaction, leading to the model $\Theta_{ij} = \alpha_i + \beta_j + u_i' v_j$. In matrix form, we can write the constraint on Θ as $\Theta = \alpha 1_m' + 1_n \beta' + UV'$, or equivalently that

$$\begin{aligned} &\min_{\Theta,\alpha,\beta,\Gamma} \mathcal{L}(\Theta; Y) + \lambda \|\Gamma\|_* \\ &\text{s.t.} \quad \Theta = \alpha 1_m' + 1_n \beta' + \Gamma. \end{aligned} \tag{21}$$

We can eliminate α and β from the problem using a similar argument as in the last section to rewrite Problem (19) as (20): for any solution (α, β, Γ) with $1_n' \Gamma \neq 0$, the alternate solution $(\alpha, \beta + n^{-1}\Gamma' 1_n, \Pi_{1_n}^{\perp}\Gamma)$ leads to the same loss but a smaller nuclear norm for Γ, since $\Gamma' \Pi_{1_n}^{\perp} \Gamma' \preceq \Gamma'\Gamma$ in semidefinite ordering. Making a similar argument for α, we can rewrite Problem (21) as

$$\min_{\Theta} \mathcal{L}(\Theta; Y) + \lambda \big\| \Pi_{1_n}^{\perp} \Theta \Pi_{1_m}^{\perp} \big\|_*, \tag{22}$$

leading to a well-defined convex optimization problem. The positive-semidefinite matrices $P = \Pi_{1_n}^{\perp}$ and $Q = \Pi_{1_m}^{\perp}$ in the penalty encode our decision to include α_i and β_j as free parameters; if $\Theta = UV'$ then U and V are only penalized insofar as they deviate from constants.

The reduction outlined above, as far as we can tell, has not explicitly appeared before—we do this to set the stage for the more general case with side information in Section 2.2.2. Note that nothing about the reduction above required any assumption on the form of $\mathcal{L}$. Hence, we can use the same reduction with an entry-wise exponential family likelihood, or Huber or absolute deviation loss; as long as $\Theta \mapsto \mathcal{L}(\Theta; Y)$ is convex, the resultant problem is convex.

2.2.2 *Features and reduced-rank vector GLMs*. Extending the previous model, we might choose to model the row effects as a linear function of the row-feature matrix $X \in \mathbb{R}^{n \times d}$. One option already discussed is to penalize only $\Pi_X^{\perp}\Theta$, imposing the model[3] $\Theta_{ij} = \alpha_j + x_i' \beta_j + \Gamma_{ij}$ where only the saturated interaction matrix Γ is penalized. If $B = [\beta_1 \cdots \beta_m] \in \mathbb{R}^{d \times m}$, this model leads to the criterion

$$\begin{aligned} &\min_{\Theta,\alpha,B,\Gamma} \mathcal{L}(\Theta; Y) + \lambda \|\Gamma\|_* \\ &\text{s.t.} \quad \Theta = 1_n \alpha' + XB + \Gamma, \end{aligned} \tag{23}$$

which in condensed form is

$$\min_{\Theta} \mathcal{L}(\Theta; Y) + \lambda \big\| \Pi_{[1,X]}^{\perp} \Theta \big\|_*. \tag{24}$$

As above, Γ is then penalized insofar as it is not explained by the features X.

A second option is to constrain $\Gamma = XB$ while using nuclear-norm regularization to enforce that B is (approximately) low-rank, leading to a reduced-rank vector GLM. If we still allow for an unpenalized intercept α_j, we could write the problem as

$$\min_{\Theta} \mathcal{L}(\Pi_{[1_n,X]}\Theta; Y) + \lambda \big\| \Pi_{1_n}^{\perp} \Theta \big\|_*, \tag{25}$$

which is always minimized by some Θ for which $\Pi_{[1,X]}\Theta = \Theta$ (otherwise $\Pi_X \Theta$ would give a smaller nuclear norm without changing the loss). We note that Yuan et al. (2007) use a nuclear norm proxy in place of the rank constraint in reduced rank regression with the least squares loss.

Note that if we allow the matrices P and Q to have some infinite eigenvalues, then (abusing notation) we could alternatively write

$$\min_{\Theta} \mathcal{L}(\Theta; Y) + \lambda \big\| (\Pi_{1_n}^{\perp} + \infty \Pi_{[1,X]}^{\perp}) \Theta \big\|_*, \tag{26}$$

where the infinite eigenvalues mean only that Θ is completely disallowed from varying in that direction [more precisely we can imagine Problem (26) as a limit of problems with $C\Pi_{[1,X]}^{\perp}$ replacing $\infty \Pi_{[1,X]}^{\perp}$, $C \to \infty$]. Thus, intercepts are unpenalized, the other directions in the span of X are penalized equally, and directions outside the span of $[1, X]$ are completely killed.

The two solutions discussed above are quite different but both enforce multitiered regularization among different left-directions of Θ. One can imagine several types of penalization schemes obtained by prioritizing left- and right-directions in the same way.

2.2.3 *Priors on latent factors and MAP estimation*. By interpreting the nuclear norm penalty as a ridge penalty on latent factors (Section 1.2.1), we reformulate MAP estimation in a variety of interesting Bayesian models as convex optimization problems. If $\Sigma \in \mathbb{R}^{n \times n}$ and $\Phi \in \mathbb{R}^{m \times m}$ are positive-definite covariance matrices reflecting correlations between rows of

[3] Aggarwal and Chen (2009) discuss similar models in the context of Bayesian hierarchical modeling—see Section 2.2.4.

the latent-factor matrices U and V, we can impose the Bayesian model:

$$\begin{aligned} &U_k \overset{\text{i.i.d.}}{\sim} N_n(0,\Sigma), \quad V_k \overset{\text{i.i.d.}}{\sim} N_m(0,\Phi), \\ &\qquad k=1,\dots,r, \\ &Y_{ij} \mid u_i, v_j \overset{\text{ind.}}{\sim} N\big(u_i' v_j, \tau^2\big), \quad (i,j)\in\Omega, \end{aligned} \tag{27}$$

where U_k is the kth column of U and u_i is the ith row. Up to a constant shift, the negative log-posterior is

$$\begin{aligned} &\sum_{(i,j)\in\Omega} \frac{1}{2\tau^2}\big(Y_{ij}-u_i' v_j\big)^2 \\ &\qquad + \sum_k \frac{1}{2} U_k' \Sigma^{-1} U_k + \sum_k \frac{1}{2} V_k' \Phi^{-1} V_k \end{aligned} \tag{28}$$

$$\begin{aligned} &= \mathcal{L}\big(UV'; Y\big) \\ &\qquad + \frac{1}{2}\big\|\Sigma^{-1/2} U\big\|_F^2 + \frac{1}{2}\big\|\Phi^{-1/2} V\big\|_F^2, \end{aligned} \tag{29}$$

where $\mathcal{L}(UV'; Y) = \sum_{(i,j)\in\Omega} \frac{1}{2\tau^2}(Y_{ij} - u_i' v_j)^2$. Using Proposition 1, we can rewrite (29) as

$$\mathcal{L}(\Theta; Y) + \big\|\Sigma^{-1/2}\Theta\Phi^{-1/2}\big\|_*. \tag{30}$$

As always, this reduction is equally correct if we replace the Gaussian log-likelihood for Y with any other log-likelihood for Y given UV'. As long as the negative log-likelihood is convex in UV', we obtain a convex problem in terms of Θ (as long as the rank of U, V is sufficiently large).

2.2.4 *MAP estimation for hierarchical priors*. To incorporate more domain knowledge or side information, various authors have proposed more complex hierarchical Bayes models which they estimate using various specialized approaches. A general modeling framework to tackle complex problems arising in recommender systems where we observe covariates $x_i \in \mathbb{R}^{d_x}$ for user i, $z_j \in \mathbb{R}^{d_z}$ for movie j, and dyadic covariates $w_{ij} \in \mathbb{R}^{d_w}$ for the pair (i,j) (for example, how many times the user has watched the movie). Following the approach of Aggarwal and Chen (2009), we can propose the more flexible generative model

$$\begin{aligned} &\eta_k \overset{\text{i.i.d.}}{\sim} N_{d_x}(0,\Sigma), \quad \zeta_k \overset{\text{i.i.d.}}{\sim} N_{d_z}(0,\Phi), \\ &\qquad k=1,\dots,r, \\ &U_{ik} \mid \eta_k \overset{\text{ind.}}{\sim} N\big(x_i'\eta_k, \sigma^2\big), \quad V_{jk} \mid \zeta_k \overset{\text{ind.}}{\sim} N\big(z_j'\zeta_k, \sigma^2\big), \\ &\qquad k=1,\dots,r, \\ &\Theta_{ij}(\alpha,\beta,\nu,X,Z,W,U,V) \\ &\qquad = \alpha' x_i + \beta' z_j + \nu' w_{ij} + u_i' v_j, \\ &Y_{ij} \mid \Theta_{ij} \overset{\text{ind.}}{\sim} \pi_{\Theta_{ij}}(y), \quad (i,j)\in\Omega. \end{aligned} \tag{31}$$

In the above model, $\pi_\theta(y)$ represents some model with convex (negative) log-likelihood such as a Gaussian, other exponential family, or log-concave location family. We can also impose a log-concave prior on $(\alpha,\beta,\nu) \in \mathbb{R}^{d_1+d_2+d_3}$ without really increasing the difficulty of the problem, but we treat them as fixed effects for simplicity. Even if (31) is a Bayesian model, it might be more practical to study nonlinear optimization methods to perform MAP estimation, as compared to using Monte Carlo methods to compute the posterior mean—in a similar vein, Aggarwal and Chen (2009) use a Monte Carlo EM procedure to compute an approximate MAP.

We take a different route—we explore when MAP estimation for the above model can be interpreted as a generalized nuclear norm regularized problem; and is hence amenable to computation via convex optimization techniques.

To see why the generalized nuclear norm framework is flexible enough to handle MAP estimation even in this complex model; observe that (up to a constant shift) the negative log-posterior is

$$\begin{aligned} &\mathcal{L}(\Theta; Y) + \frac{1}{2\sigma^2}\|U - X\eta\|_F^2 + \frac{1}{2\sigma^2}\|V - Z\zeta\|_F^2 \\ &\qquad + \frac{1}{2}\big\|\Sigma^{-1/2}\eta\big\|_F^2 + \frac{1}{2}\big\|\Phi^{-1/2}\zeta\big\|_F^2, \end{aligned} \tag{32}$$

where we have suppressed the dependence of Θ on the other variables.

The function (32) may appear daunting at first blush due to many nonconvex bilinear terms. However, by partially minimizing with respect to η and ζ we can massage it into a friendlier form. We first write

$$\begin{aligned} &\frac{1}{2\sigma^2}\|U - X\eta\|_F^2 + \frac{1}{2}\big\|\Sigma_\eta^{-1/2}\eta\big\|_F^2 \\ &\qquad = \sum_k \bigg(\frac{1}{2\sigma^2}\|U_k - X\eta_k\|_2^2 + \frac{1}{2}\big\|\Sigma_\eta^{-1/2}\eta_k\big\|_2^2 \bigg), \end{aligned}$$

which is a separable sum of generalized ridge regression criteria, each regressing U_k against X. For any fixed U_k, the kth term is minimized by setting $\eta_k = (X'X + \sigma^2\Sigma^{-1})^{-1}X'U_k$. Substituting back into the original expression and simplifying, we obtain

$$\begin{aligned} &\min_{\eta_k} \frac{1}{2\sigma^2}\|U_k - X\eta_k\|_2^2 + \frac{1}{2}\big\|\Sigma_\eta^{-1/2}\eta_k\big\|_2^2 \\ &\qquad = \frac{1}{2\sigma^2} U_k'(I-H)U_k, \end{aligned}$$

where $H = X(X'X + \sigma^2\Sigma^{-1})^{-1}X'$. After eliminating ζ the same way, we obtain the criterion:

$$\mathcal{L}(\Theta; Y) + \frac{1}{2\sigma^2}\|(\mathbb{I} - H)U\|_F^2 + \frac{1}{2\sigma^2}\|(\mathbb{I} - G)V\|_F^2, \tag{33}$$

where $G = Z(Z'Z + \sigma^2\Phi^{-1})^{-1}Z'$. In light of Proposition 1, the above leads to the equivalent minimization problem

$$\begin{aligned} &\mathcal{L}(\Theta; Y) + \frac{1}{\sigma^2}\|(\mathbb{I} - H)\Gamma(\mathbb{I} - G)\|_* \\ &\text{s.t.} \quad \Theta = X\alpha + \beta'Z' + \langle \nu, W\rangle + \Gamma, \end{aligned} \tag{34}$$

if $r \geq \operatorname{rank}(\hat{\Gamma})$, where $\hat{\Gamma}$ is a solution to Problem (34);[4] and we write $\langle \nu, W\rangle = (\nu' w_{ij} : i \leq n, j \leq m)$. The convex optimization formulation (34) provides a new perspective of the highly nonconvex MAP estimation task implied by the model (31); and is novel, to the best of our knowledge.

Equation (34) shows that we are now only penalizing the *residuals* of U_k and V_k relative to the (ridge-penalized) linear models in X and Z. If desired, we can further eliminate the variables α and β as discussed in previous sections.

3. MISSING DATA AND LEARNING

In most of the matrix completion literature, the missingness pattern Ω is implicitly assumed to be uninformative. However, in many of the most salient applications for matrix completion and low-rank modeling, missingness is highly informative. For example, in the case of Netflix data, it is highly implausible that a user chooses movies to watch without any regard to whether they anticipate enjoying those movies. As a result, *which movies the users choose to rate* can provide a great deal of insight into their latent types, one of the key insights driving the prize-winning algorithm (Bell and Koren, 2007). We will use the Netflix problem as a running example in this section.

In matrix completion problems, the row and column identities play a very different role than they do in more typical data matrices where each row represents an independent observational unit. Though we abstractly represent these identities by the indices i and j, they are not merely anonymous replicates: for example, in the Netflix data they correspond to the identities of the individual users and movies about which we are interested in learning. Thus, we consider each entry of the matrix Y to be an observational unit, with possibly unobserved response Y_{ij} and observed predictor variables given by the observed row and column identities i and j, as well as any other side information relating to the row, column or entry. Viewed in this way, the parameter matrix $\Theta_{ij} = f(i, j)$ is a regression function mapping the predictors i and j to determine the conditional distribution of the response Y_{ij}, and this mapping is parameterized by quantities such as α_i, β_j, u_i, and v_j, which we view as fixed parameters.

Following convention in the missing-value literature, we can introduce Bernoulli indicator variables for the missingness pattern $M_{ij} = 1\{(i, j) \notin \Omega\}$. For simplicity, we will assume throughout that (M_{ij}, Y_{ij}) pairs are independent of each other, with

$$\Xi_{ij} = \log \frac{\mathbb{P}((i, j) \in \Omega)}{\mathbb{P}((i, j) \notin \Omega)}, \tag{35}$$

or equivalently $\mathbb{E}M_{ij} = (1 + \exp\{\Xi_{ij}\})^{-1}$. The data are missing completely at random, then, if M_{ij} is completely independent of i, j, and Y_{ij}—that is, if every entry is equally likely to be observed. This scenario seems highly unlikely for Netflix data as well as most other matrix completion problems.

3.1 Data Missing at Random

The missing data are ignorable in this case if and only if Y_{ij} is independent of M_{ij} given the categorical row and column predictors i and j—that is, in terms of the framework of Rubin (1976), whether the data are *missing at random* (MAR). For example, in the Netflix data, a user may preferentially watch movies that she expects to align with her preferences, but once she decides to watch a movie her decision of whether to rate it is unrelated to her evaluation of the movie.

If the data are MAR in the sense above, then the conditional likelihood of Y_{ij} given $M_i = 1$ is the same as the conditional likelihood of Y_{ij} given $M_i = 0$. Therefore, we can still learn to predict the missing cases by analyzing the non-missing cases. However, as Bell and Koren (2007) found, this may be a highly suboptimal, especially if Ξ_{ij} is partly driven by the same parameters α_i, β_j, u_i, and v_j that determine Θ_{ij}.

Viewed this way, missingness at random is a special case of the multi-task learning problem, simply adding more data for estimating the same parameters, possibly in addition to some more parameters. For example, we might add an extra parameter ρ_i for user i,

[4] We note that if $r < \operatorname{rank}(\hat{\Gamma})$ then this equivalence does not hold and a solution to Problem (33) may not be obtained via convex optimization.

parameterizing her propensity to watch more movies, and τ_j parameterizing movie j's overall prevalence, and model $\Xi_{ij} = \rho_i + \tau_j + u_i v_j'$ or equivalently $\Xi = \rho 1' + 1\tau' + UV'$. We could fit this model with minimal modification to the algorithmic framework described below

Alternatively, we might believe the user/movie interaction should be similar but not exactly the same for Ξ and Θ. Then, we could model

$$\Xi_{ij} = \rho_i + \tau_j + r_i t_j',$$

and penalize $\lambda_{ru}\|r_i - u_i\|_2^2$ and $\lambda_{tv}\|t_j - v_j\|_2^2$. Mapping this back to a matrix completion problem, we would arrive at partially missing data matrix $\widetilde{Y}$, and parameter matrix $\widetilde{\Theta}$, where

$$\widetilde{Y} = \begin{bmatrix} Y & — \\ — & M \end{bmatrix}, \quad \widetilde{\Theta} = \widetilde{U}\widetilde{V}' = \begin{bmatrix} U \\ R \end{bmatrix} \begin{bmatrix} V' & T' \end{bmatrix}.$$

The character "—" denotes completely missing blocks of $\widetilde{Y}$ that play no role in the likelihood; note this means that the values of UT' and RV' do not matter. Finally, we would be left with the penalized likelihood criterion

$$\mathcal{L}(\widetilde{\Theta}; \widetilde{Y}) + \lambda_{ru}\|U - R\|_F^2 + \lambda_{tv}\|T - V\|_F^2,$$

where the quadratic penalties can be rewritten as $\|U - R\|_F^2 = \|[1_n 1_n' - (1_n 1_n')]\widetilde{U}\|_F^2$.

More generally, we can imagine an infinitude of possible ways of modeling the missing data pattern, many of which fit quite neatly into the computational framework described here.

3.2 Data Missing Not at Random

By contrast, we might imagine that another user rates movies strategically, only bothering to rate those movies he especially likes or dislikes, or searching his memory to rate his favorite movies that he watched long ago. In that case, the movies are *missing not at random* (MNAR), the most general and least favorable scenario. If the missing mechanism is MNAR then, even if we successfully learn to predict Y_{ij} for observed entries $(i, j) \in \Omega$, we cannot necessarily rely on our predictions to perform well for values of Y_{ij} for the not-yet-observed entries $(i, j) \notin \Omega$. From the perspective of a firm like Netflix, this may pose a major problem, especially if they aim to use the data to recommend movies that a user has not yet rated, but which they believe he would like.

As in most problems, it is impossible to determine from the data alone whether the data are MAR or MNAR (or to correct for MNAR data). To determine whether (or how badly) the data are MNAR, we could however use auxiliary data. For example, we could imagine that Netflix keeps data about (a) which movies a user watched on Netflix's recommendation, (b) which movies he watched by his own choice (e.g., by searching for them), and (c) which movies he rated without watching on Netflix. Then Netflix could train a model on ratings in groups (b) and (c), and test for group (a); if the predictions are unsuccessful, Netflix could modify its model to take this into account—for example, by introducing a categorical predictor variable W_{ij} encoding a, b, c, or d if the rating is still missing, and modeling the way that W_{ij} influences Y_{ij}.

4. COMPUTATION

We now review several computational methods that can be used to solve optimization problems with generalized nuclear norm regularization. Many of these methods have been used successfully to address the special case of matrix completion with nuclear norm regularization and closely related problems. We present an overview of these methods highlighting their advantages and disadvantages. We then discuss how these ideas may generalize to the class of problems we discuss in this paper.

4.1 Algorithms for Solving the Nuclear Norm Regularized Problem

In the discussion below, with a slight abuse of notation, we will denote $\mathcal{L}(\Theta; Y)$ by $\mathcal{L}(\Theta)$. We will assume that $\mathcal{L}(\Theta)$ in Problem (4) is convex and differentiable. These methods also extend to a fairly large family of non-smooth functions by applying Nesterov's smoothing technique (Nesterov, 2005).

4.1.1 *Proximal gradient algorithms*. We first discuss obtaining approximate solutions to Problem (4), that is, we are interested in minimizing the function $H(\Theta) := \mathcal{L}(\Theta) + \lambda\|\Theta\|_*$ w.r.t. Θ. If $\mathcal{L}(\Theta)$ is differentiable with Lipschitz continuous gradient[5]

$$\|\nabla\mathcal{L}(A) - \mathcal{L}(B)\|_F \le L\|A - B\|_F, \quad \forall A, B \in \mathbb{R}^{n\times m}, \tag{36}$$

[5]We note that for the Poisson distribution, $\mathcal{L}(\cdot)$ does not satisfy this property. However, if an optimal solution is bounded (which is typically the case), it will *effectively* satisfy (36) with $L < \infty$—see, for example, Atchadé, Mazumder and Chen (2015) for a formal treatment of this aspect, in the context of the graphical lasso problem.

then proximal gradient methods (Beck and Teboulle, 2009) can be used, leading to the updates

$$\begin{aligned}\Theta_{k+1} \in \operatorname*{argmin}_{\Theta} & \frac{L}{2}\left\|\Theta - \left(\Theta_k - \frac{1}{L}\nabla\mathcal{L}(\Theta_k)\right)\right\|_F^2 \\ & + \lambda\|\Theta\|_* \\ = S_{\lambda/L}&\left(\Theta_k - \frac{1}{L}\nabla\mathcal{L}(\Theta_k)\right),\end{aligned} \tag{37}$$

where $S_\tau(A)$ is defined in (5). It follows from the convergence properties of proximal gradient methods that $H(\Theta_k) \to \min_\Theta H(\Theta)$ as $k \to \infty$; and the convergence rate is sublinear, that is, after k many iterations, $H(\Theta_k)$ is at most $O(1/k)$-away from the minimum of $H(\Theta)$ (Beck and Teboulle, 2009). When the loss function is taken as the squared error over the observed entries Ω, and $L = 1$, this is the *Soft-Impute* algorithm of Mazumder, Hastie and Tibshirani (2010). We note that accelerated gradient methods (Beck and Teboulle, 2009) can also be used, which have a (superior) convergence rate of $O(1/k^2)$ as compared to the sublinear rate of proximal gradient methods.[6] Every iteration requires computing the singular value thresholding operator, which can be done fairly quickly via a full SVD if the problem size is small (few hundred rows/columns). If the matrices are larger, computing (37) becomes more involved; and specialized numerical linear algebra routines for low-rank SVDs are needed. Note that if λ is large then the number of nonzero singular values in $S_\lambda(A)$ is small—it therefore suffices to compute a low-rank SVD of A, which can be significantly cheaper than computing a full SVD of A with cost $O(m^2n)$ (assuming that $n \geq m$). If A has special structure (for example, a sparse matrix) for which matrix-vector multiplications of the form Ab_1 and $A'b_2$ are cheap, then an approximate low-rank SVD can be computed with several such matrix-vector multiplications. The popular power method (Golub and Van Loan, 1983) is often used to compute the largest singular-vector/value of large matrices. The block QR method or alternating least squares (Golub and Van Loan, 1983) (see also the *Soft-Impute* package of Hastie et al., 2015) method and algorithms based on Lanczos subspace iterations (Golub and Van Loan, 1983) as implemented in the PROPACK software (Larsen, 2004) are extremely effective methods for computing the top few singular values and vectors for large matrices for which matrix-vector multiplications are cheap. Below we provide some examples wherein A is structured—this enables fast multiplications of A and A' with a vector.

Matrix completion: For the matrix completion problem with least squares loss, we have

$$\mathcal{L}(\Theta) = \frac{1}{2}\sum_{(i,j)\in\Omega}(\theta_{ij} - y_{ij})^2 = \frac{1}{2}\|P_\Omega(\Theta - Y)\|_F^2,$$

where $P_\Omega(\Theta)$ is the projection matrix onto the observed entries, that is, the (i, j)th entry of $P_\Omega(\Theta)$ is θ_{ij} if $(i, j) \in \Omega$ and zero otherwise. We let $P_\Omega^\perp(\Theta) = \Theta - P_\Omega(\Theta)$. $P_\Omega(\Theta)$ is a sparse matrix with atmost $|\Omega|$ many nonzeros, which can be potentially much smaller than mn. For the Netflix dataset for example, $|\Omega|/mn$ is 1.2%. Update (37) for this problem entails computing a low-rank SVD of the matrix $P_\Omega(Y) + P_\Omega^\perp(\Theta_k)$, which, curiously can be written as the sum of a sparse and low-rank matrix:

$$\begin{aligned}\widetilde{\Theta}_k &:= P_\Omega(Y) + P_\Omega^\perp(\Theta_k) \\ &= \underbrace{P_\Omega(Y - \Theta_k)}_{\text{Sparse}} + \underbrace{\Theta_k}_{\text{Low rank}},\end{aligned} \tag{38}$$

wherein Θ_k is anticipated to be of low-rank since a sufficiently large value of λ in Problem (4) encourages a low-rank solution. In practice, one maintains an upper bound $\bar{r}$ on the maximum allowable rank on Θ_k for improved memory and storage usage (Mazumder, Hastie and Tibshirani, 2010). In addition, we never need to store or form the entire matrix Θ_k. Instead, we store factors (A_k, B_k) where $\Theta_k = A_k B_k'$ and this stems from the low-rank SVD of Θ_k. Suppose $\tilde{r}$ is the "working" rank of Θ_k, that is, A_k, B_k have $\tilde{r}$ columns each. We note that computing $P_\Omega(Y - \Theta_k) = P_\Omega(Y) - P_\Omega(\Theta_k)$ requires evaluating the entries of Θ_k for all $(i, j) \in \Omega$ which can be done with cost $O(|\Omega|\tilde{r})$ by using the factored representation of Θ_k. Note that multiplying $\widetilde{\Theta}_k$ with a vector is of cost $O(|\Omega|) + O((m + n)\tilde{r})$. Usually in matrix completion problems, we seek $\tilde{r}$ latent factors with $\tilde{r} \ll m, n$; and $|\Omega|$ is comparable to $O((m + n)\tilde{r})$—thus the matrix vector multiplications are of cost $O((m + n)\tilde{r})$. When $|\Omega|$ is small, the above techniques also generalize to loss functions corresponding to other members of the generalized linear model family as long as (36) holds true.

Structured gradients: Let us consider some other loss functions $\mathcal{L}(\Theta)$ arising in problems that are different from missing data/matrix completion problems

[6] Accelerated gradient methods should be used with caution for large problems, especially with inexact computation of the proximal steps. The latter arises from approximate low-rank SVD computations and may lead to possible unstable behavior and nonconvergence of the algorithm due to error accumulation (Devolder, Glineur and Nesterov, 2014).

discussed above. To efficiently compute (37), we will need to efficiently compute the gradient of the loss function $\mathcal{L}(\Theta)$ w.r.t. Θ and also perform fast matrix-vector multiplications of the form: $\nabla\mathcal{L}(\Theta)b_1$ and $\nabla\mathcal{L}(\Theta)'b_2$. For a problem of the form (11), the method of Section 4.1.1 requires us to compute the gradient of the smooth mapping $\Theta \mapsto \mathcal{L}(\tilde{P}\Theta\tilde{Q})$ w.r.t. Θ. Writing $\Theta = AB'$ (assuming that A, B have a small working rank) the gradient is given by $\nabla_\Theta\mathcal{L}(\tilde{P}\Theta\tilde{Q}) = \tilde{P}\nabla_Z\mathcal{L}(Z)\tilde{Q}$ with $Z = \tilde{P}\Theta\tilde{Q}$. A key to efficiently computing (37) requires performing fast matrix-vector multiplications of the form: $\tilde{P}\nabla_Z\mathcal{L}(Z)\tilde{Q}b_1$ [and also $(\tilde{P}\nabla_Z\mathcal{L}(Z)\tilde{Q})'b_2$]. This is easy to compute as long as: Multiplying $\tilde{Q}$ and $\tilde{P}$ with a vector is computationally cheap. This is the case when the matrices $\tilde{P}, \tilde{Q}$ are sparse, low-rank or the sum of a sparse and low-rank matrix (for example). All examples discussed in the paper including the ones in Section 5 satisfy this property.

For large problems, that is, when the number of rows/columns become a few thousand (say, 5000 or larger), the computational cost of these iterative procedures relying on matrix-vector multiplications, will increase substantially if the gradients do not have sufficient structure (similar to that described above).

4.1.2 *Nonconvex optimization algorithms*. Another class of algorithms that are commonly used for matrix completion problems exploit the equivalence between Problems (7) and (4). Herein, we attempt to directly optimize the nonconvex objective $\mathcal{L}(UV') + \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2$ in terms of the variables $(U_{n\times r}, V_{m\times r})$. Note that we need to consider a value of r that is large enough to ensure that Problems (7) and (4) are equivalent. We acknowledge that choosing this value of r is difficult, as the rank of $\hat{\Theta}$, a minimizer of Problem (4) is not known in advance. However, this can be estimated during the course of the algorithm, as we discuss below.

Rennie and Srebro (2005) propose using gradient descent on Problem (7) for the matrix completion problem. Hastie et al., 2015 use an inexact block coordinate stylized method for the matrix completion problem, motivated by the EM-algorithm underlying *Soft-Impute* (Mazumder, Hastie and Tibshirani, 2010). While nonconvex problems are prone to local minima and saddle points, it turns out that under certain additional verifications/checks (Burer and Monteiro, 2005, Hastie et al., 2015, Journée et al., 2010), these nonconvex algorithms lead to solutions of the convex optimization problem (4). We note however, that certifying whether a pair (U, V) is a local minimizer requires checking the positive semi-definiteness of a Hessian operator (Journée et al., 2010), which may be difficult to verify for large scale problems. Hastie et al. (2015) show that a singular value thresholding operation, inspired by *Soft-Impute*, can be performed to check if a stationary point of (7) corresponds to the global minimizer of the convex nuclear norm regularized Problem (4). We refer the reader to the work of Hastie et al. (2015) for a detailed investigation of these issues for the matrix completion problem, wherein the authors also show that nonconvex algorithms for the matrix completion problem can be much more efficient than usual proximal gradient type methods for the problem. We note that a key requirement in this approach is the choice of r, if r is smaller than $\hat{r}$ then Problem (7) will not be equivalent to Problem (4); if r is taken to be too large then this nonconvex optimization approach (in the U, V variables) will be computationally expensive. A practical strategy is to (i) start with a small value of r and optimize Problem (7) with respect to (U, V) to get a stationary point; (ii) check if the conditions of optimality with respect to the convex problem are met; (iii) if the conditions are not satisfied, then one can increase r and repeat the optimization process with warm-starts (Hastie et al., 2015).

4.1.3 *Frank–Wolfe type algorithms*. Fairly recently, a class of first order methods known as Frank–Wolfe aka Conditional Gradient algorithms have gained popularity in the context of nuclear norm regularized problems; and in particular, the problem of matrix completion with the squared error loss function. We refer the reader to an incomplete list of papers by Frank and Wolfe (1956), Jaggi and Sulovsk (2010), Freund, Grigas and Mazumder (2017) (see also references therein) that have pursued this line of investigation. The Frank–Wolfe algorithm operates on the constrained version of the nuclear norm regularized problem (3) given by: $\min\{\mathcal{L}(\Theta) : \|\Theta\|_* \le \tau\}$. Note that we will assume that condition (36) holds true. A particularly appealing aspect of this algorithm is that at every iteration it computes a rank-one SVD of a $n \times m$ matrix.

The Frank–Wolfe algorithm gives rise to the update sequence:

$$\begin{aligned}&\Theta_{k+1} = \Theta_k + \alpha_k(\tilde{\Theta}_{k+1} - \Theta_k) \quad \text{where}\\ &\tilde{\Theta}_{k+1} \in \operatorname{argmin}\big\{\langle\nabla\mathcal{L}(\Theta_k), \Theta\rangle : \|\Theta\|_* \le \tau\big\}\end{aligned} \tag{39}$$

for a sequence $\alpha_k = 2/(k+2)$, where k denotes the iteration counter. This sequence $\mathcal{L}(\Theta_k)$ converges to the optimum of Problem (3) with a finite time convergence rate of $O(1/k)$. An appealing trait of this algorithm is

that $\tilde{\Theta}_{k+1}$ [as in (39)] requires computing the largest singular vector/value of the matrix $\nabla\mathcal{L}(\Theta_k)$ which can be done via the power-method, for example. For matrix completion with the squared error loss, the gradient of the loss function is: $\nabla\mathcal{L}(\Theta_k) = P_\Omega(Y - \Theta_k)$ which is a sparse matrix with $O(|\Omega|)$ nonzero entries and hence a power method will entail a per-iteration cost of $O(|\Omega|)$. When $\mathcal{L}(\Theta_k)$ has no specialized structure, computing $\tilde{\Theta}_{k+1}$ can be achieved via the power method with cost $O(mn)$—this is usually much cheaper than computing a thresholded SVD as in (37). The caveat of the Frank–Wolfe method is that it can take several iterations to reach an approximate solution to Problem (3) with a small rank, even if we assume that the problem admits a low-rank solution. If τ is taken such that $\hat{\Theta}$, an optimal solution to Problem (3), has a rank of 20 (say), then $\text{rank}(\Theta_k)$ can quite easily become of the order of a thousand with as many iterations. As the number of iterations increase and Θ_k makes its way to an optimal solution, the rank gradually decreases. This is in contrast to proximal gradient methods (Mazumder, Hastie and Tibshirani, 2010, Hastie et al., 2015) where the nuclear norm thresholding operator induces a low-rank solution via the soft-thresholding operation on the singular values. There are sophisticated variants of the Frank–Wolfe method with "In face" extensions that can address these shortcomings, with marginally more computational cost—we refer the reader to Freund, Grigas and Mazumder (2017) and references therein for an in-depth investigation.

For general problems where $\nabla\mathcal{L}(\Theta_k)$ is not sparse or sufficiently structured, pre or post-multiplying $\nabla\mathcal{L}(\Theta_k)$ with a vector will cost $O(mn)$; thereby, computing $\tilde{\Theta}_{k+1}$ for many iterations will become computationally expensive. Since the vanilla version of the Frank–Wolfe method does not necessarily lead to low-rank solutions along the course of the algorithm, storage/memory constraints will limit storing Θ_k as soon as k becomes sufficiently large.

4.1.4 *Other algorithms*. Another approach for low-rank models directly operate on the rank constrained optimization problem (1) in terms of the latent factors U, V. In this approach, we do not consider the regularization term as discussed in Section 4.1.2. If $\Theta \mapsto \mathcal{L}(\Theta)$ is convex, then $\mathcal{L}(UV')$ is convex in V (for fixed U); and vice-versa. One can apply alternating minimization or block coordinate descent (Bertsekas, 1999) algorithms for this problem. However, as explained in Section 4.1.2 the algorithm may lead to undesirable stationary points or saddle points. Additional assumptions on the problem data are needed to make the algorithm more well behaved. Toward this end, Jain, Netrapalli and Sanghavi (2013) analyze the behavior of such algorithms in the context of matrix completion problems with squared error loss under incoherence-like assumptions on the underlying data generating mechanism. This is similar in spirit to conditions required for nuclear norm regularized matrix completion (Candès and Tao, 2010) to recover an underlying low-rank matrix. Chen and Wainwright (2015) study the behavior of these algorithms for more general loss functions. The quality of the solutions produced by these algorithms when such assumptions are violated, however, is not clear. It is useful to remind ourselves that the nuclear norm regularization provides shrinkage on the singular values of the matrix; and this may lead to better generalization error. This may be a reason why criterion (3) may be preferred over the (unregularized) rank constrained version (1). We refer the reader to the recent work of Mazumder, Radchenko and Dedieu (2017) for similar discussions in the context of high dimensional sparse linear regression.

Approximate message passing: Another appealing approach to low-rank matrix completion is based on the Approximate Message Passing (AMP) algorithmic framework. These class of algorithms have been inspired by the success of similar algorithms in the compressed sensing literature; and have been studied by a series of recent works (Parker, Schniter and Cevher, 2014a, 2014b, Lesieur, Krzakala and Zdeborová, 2015). In the context of matrix completion, Parker, Schniter and Cevher (2014b) show that their proposed algorithm leads to superior reconstruction error and also faster runtimes when compared to other off-the-shelf algorithms for matrix completion, and related problems. Understanding deeper statistical and computational connections between these algorithms and other methods described herein (see for example, Section 2.2) might be an interesting direction for future research. It may also be interesting to explore if AMP based methods can be extended to handle side-information as discussed in Section 2.2.

4.2 Solving the Generalized Nuclear Norm Problem

Motivated by Problem (8), we discuss techniques to minimize $G(\Theta) := \mathcal{L}(\Theta; Y) + \lambda\|P\Theta Q\|_*$, for different choices of P, Q.

If P, Q are invertible, $G(\Theta)$ can be reformulated as an instance of Problem (4) upon performing a suitable change of variables; and the methods described above are applicable. We note that as long as m, n are of the order of a few thousands each, the matrix inversions for

P, Q are computationally feasible. If m, n are larger (in the order of tens of thousands, for example), it may still be computationally feasible to invert P provided it has some special structure. For example, if $P = \mathbb{I} - H$ for H low-rank, then P^{-1} (assumed to exist) can be obtained by using the Sherman Woodbury formula. A similar story applies to Q as well. In addition to being able to compute inverses of P, Q; one also needs to compute the gradient quite efficiently—this is possible as long as it is fast to multiply P^{-1}, Q^{-1} with vectors.

If at least one of P or Q is low rank, then $G(\Theta)$ can be minimized using the Alternating Direction Method of multipliers method (Boyd et al., 2011) as we discuss in Appendix B.

Finally, another approach to minimize $G(\Theta)$ is to express Θ in terms of U, V (note that we assume $\Theta = UV'$) and consider an optimization problem in the form of Problem (9). One can then directly apply nonlinear optimization methods on the problem; as discussed in Section 4.1.2. For example, one can apply gradient descent on the function with respect to latent variables $U_{n\times r}, V_{m\times r}$. A stationary point of such an algorithm will correspond to the minimum of the convex problem (8) if: r is chosen sufficiently large, and the point $\hat{\Theta} = \hat{U}\hat{V}'$ corresponds to a local minimum of the objective function. Checking the latter usually entails verifying whether $\hat{U}\hat{V}'$ satisfies the optimality condition of the corresponding convex problem.

Multiple blocks: Let us consider the general convex problem (34), where $\Theta_{ij} = X\alpha + \beta' Z' + \langle \nu, W \rangle + \Gamma$. We can apply a block coordinate descent algorithm (Bertsekas, 1999) across the blocks (α, β, ν) and Γ. The optimization problem w.r.t. (α, β, ν) is a simple convex optimization problem and is straightforward to do. The optimization problem w.r.t. Γ is a (generalized) nuclear norm regularized problem, and has been discussed above.

4.3 A Summary of the Current Computational Landscape

As we have discussed above, there are a wide number of possible algorithmic approaches for the nuclear norm regularized matrix completion problem. In our opinion, every method has its advantages and disadvantages. It is not clear to us if any one method completely dominates the rest in terms of speed, robustness, generalizability to different loss functions, scalability and computational guarantees. Due to the close ties of these convex problems to low-rank matrix decompositions, it is not surprising that many of these approaches make heavy use of techniques from numerical linear algebra such as those arising from computing low-rank SVDs. The basic versions of most of the methods described above are rather simple to implement; and are likely to lead to (more or less) similar performances on many moderate-sized instances. We do note however, that additional work may be needed for specialized and careful implementations of these algorithms to obtain improved performance in practice, especially for large scale problems. For instance, the proximal gradient methods require a knowledge of the Lipschitz constant, though a line-search (Beck and Teboulle, 2009) can be used to estimate it, if it is unknown. However, the latter may become quite expensive for large scale instances due to multiple evaluations of the proximal step which involves a singular value thresholding operation (37). On the other hand, the basic version of the Frank–Wolfe method is relatively simple to implement, and unlike proximal gradient methods, it does not require any prior knowledge of the value of the Lipschitz constant. However, to encourage low-rank solutions along the course of the algorithm, non-trivial modifications to the basic Frank–Wolfe algorithm are required. For large problems, nonconvex optimization based approaches or alternating (bi-convex) optimization procedures are perhaps the easiest to implement and require a minimal working knowledge of modern convex optimization algorithms. However, the quality of the stationary points can be quite suboptimal; and additional considerations are needed to verify the quality of the solution on arbitrary real-data instances (provided we do not make any unverifiable assumption on the problem data). For large problems, when rows/columns are of the order of tens of thousands, memory management issues become important: Anticipating a low rank solution to Θ, it is imperative that Θ be stored in factored form in terms of its latent factors. Almost all the methods mentioned above, take that into account in some form or the other.

When one considers general smooth and convex loss functions, beyond those arising in the context of matrix completion, all algorithms are likely going to be much slower, especially, for larger problem sizes. The generalized nuclear norm problem is arguably more challenging than the vanilla nuclear norm regularized problem and the current computational landscape for these problems is not as well understood as the vanilla nuclear norm regularized problem. This should perhaps not come across as too surprising, as a similar story arises in the context of usual penalized regression. It is well known that in regularized linear regression,

the two dimensional total variation denoising problem or the generalized lasso problem is computationally more challenging than the usual Lasso (Hastie, Tibshirani and Wainwright, 2015, Nesterov, 2004, Beck and Teboulle, 2009). Furthermore, nuclear norm regularization leads to semidefinite optimization problems. They are widely acknowledged to be significantly more difficult optimization problems when compared to quadratic optimization problems arising in high dimensional linear regression. Developing faster algorithms for the generalized nuclear norm regularized problem is an important direction for future research. We contend that specialized methods need to be tailored to the structure and dimensions of the matrices P, Q; and the stylized loss function.

5. DATA EXAMPLES

We now describe the results of our method on two applications, meant both to illustrate the scalability of our algorithm, and to suggest the level of generality of possible models by way of example.

5.1 Functional Data Reconstruction

Our first example is of a nonparametric flavor and involves real data. We begin with 200 noisy functions measured at 256 equally spaced frequency points. Each function is a log-periodogram computed from a 32 ms recording of a male research subject speaking one of five phonemes. The data were originally collected in the TIMIT speech corpus from the U.S. Department of Commerce and processed and analyzed in Hastie, Buja and Tibshirani (1995) to demonstrate a variant of discriminant analysis with smoothness penalty applied, with a goal of classifying the phonemes. Because each log-periodogram shows erratic variation around a smooth trend, a successful statistical analysis must extract the smooth structure without overfitting to the noise.

Each log-periodogram is a 256-dimensional vector, and we combine them to form a matrix $Y \in \mathbb{R}^{200\times 256}$, whose ith row is the ith log-periodogram. We expect the signal in this matrix to be low-rank, and the latent column factors should be smooth in frequency. We then artificially construct a sparsely-observed data set by sampling each curve at 26 random points, and set ourselves the objective of reconstructing the functions based on these relatively few samples, exploiting the assumptions of smoothness and low rank. The purpose of this example is to show that forcing the latent variables to be smooth in frequency can demonstrably improve in reconstruction accuracy.

To proceed, we impose a simple model on the column variables v_j; they are constrained to lie in the natural spline basis with 12 degrees of freedom, and we shrink them toward the natural spline basis with 4 degrees of freedom. We estimate the principal components analysis model with unpenalized marginal column (time) effects:

$$\min \mathcal{L}(\Theta; Y) + \lambda_\Gamma \left\| \Gamma^{(2)}_{ij} \right\|_* \tag{40}$$

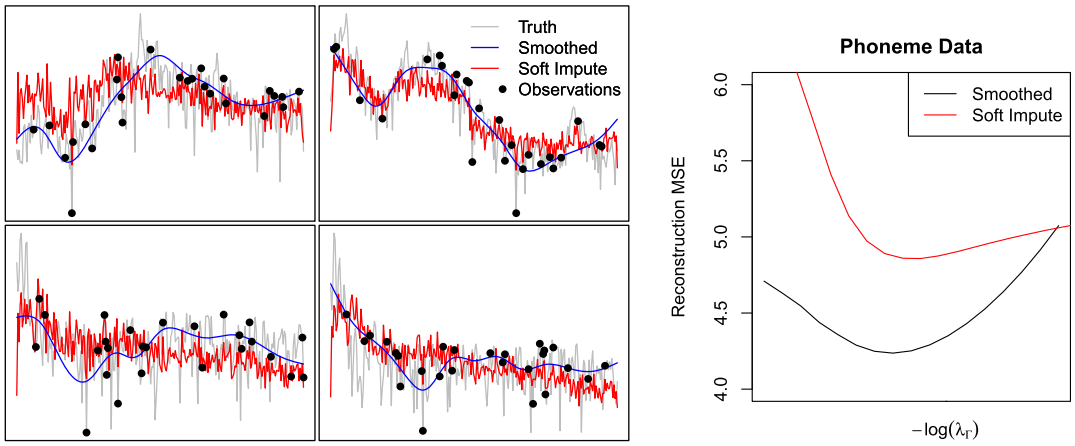
$$\text{s.t.} \quad \Theta_{ij} = \mu + \beta_j + \Gamma^{(1)}_{ij} H_4 + \Gamma^{(2)}_{ij} H_{12}, \tag{41}$$

where H_d represents a d-dimensional natural spline basis (with intercept). This can be framed (somewhat artificially) as a Bayesian prior on V with flat variance in the directions of H_4, finite positive variance in the directions of H_{12}, and zero variance on other directions. Although this data set is relatively small, it can be computationally advantageous to constrain Γ as we have done here, since it reduces the size of our optimization variables (e.g., $\Gamma^{(2)}$ is $n \times 12$ instead of $n \times 256$). We compare our proposed method to matrix completion using the standard nuclear norm, which does not exploit smoothness. The right panel of Figure 1 shows that side information cuts MSE by a sizeable fraction.

5.2 Reduced-Rank Poisson Regression for Ecological Modeling

As a second example to illustrate the richness of our modeling framework, we consider an ecological application, species distribution modeling using presence-only data. On a geographic domain $\mathcal{D}$, we observe a process of point observations for each of m species, with the goal of determining the abundance of each species as a function of geographic location, or understanding the determinants of habitat suitability. The observations typically arise from opportunistic sources such as museum collections or citizen science data, leading to a strong sampling bias toward population centers. Our model extends a model proposed in Fithian et al. (2015) by introducing low-rank regularization to borrow strength across species.

For $s \in \mathcal{D}$, let $x(s)$ denote a d-variate vector of habitat covariates that drive species abundance, and let $z(s)$ denote other covariates driving the sampling bias. We assume that species j has a latent *species process* $\mathcal{S}_j$ representing all locations where species j occurs, modeled as an inhomogeneous Poisson point process with *species intensity* $\nu_j(s) = \exp\{\alpha_j + x(s)'\beta_j\}$. The species process is then filtered through a biased observation model wherein each occurrence is observed with probability $b_j(s)$, so that we only observe the



(a) (b)

FIG. 1. *Phoneme data, comparison of* Soft-Impute (*matrix completion with the standard nuclear norm penalty*) *against a generalized nuclear norm penalty with smoothness enforced.* (a) *Reconstruction for the first four phoneme curves, with regularization parameter* λ *chosen by cross-validation.* (b) *Mean squared error for held-out data as a function of* λ.

thinned Poisson process $\mathcal{T}_j$ with intensity $v_j(s)b_j(s)$. We model $b_j(s) = \exp\{\epsilon_j + z(s)'\delta\}$ (with δ *not* varying by species), reflecting an opinion that the spatial bias is a function of observer behavior only.

Typically the geographic domain is discretized into n pixels reflecting the resolution of covariate measurements; if s represents a pixel with unit area, let Y_{sj} denote the number of $s \in \mathcal{T}_j$ falling into pixel s. Then,

$$Y_{sj} \sim \text{Pois}\big(\exp\{\alpha_j + x(s)'\beta_j + \epsilon_j + z(s)'\delta\}\big).$$

Since α_j and ϵ_j are unidentifiable in this model, the species intensity v_j is also unidentifiable. However, we can estimate the normalized *species distribution* $p_j(s) = v_j(s)/\int_{\mathcal{D}} v_j(s)$, which depends only on β_j and is therefore identifiable.

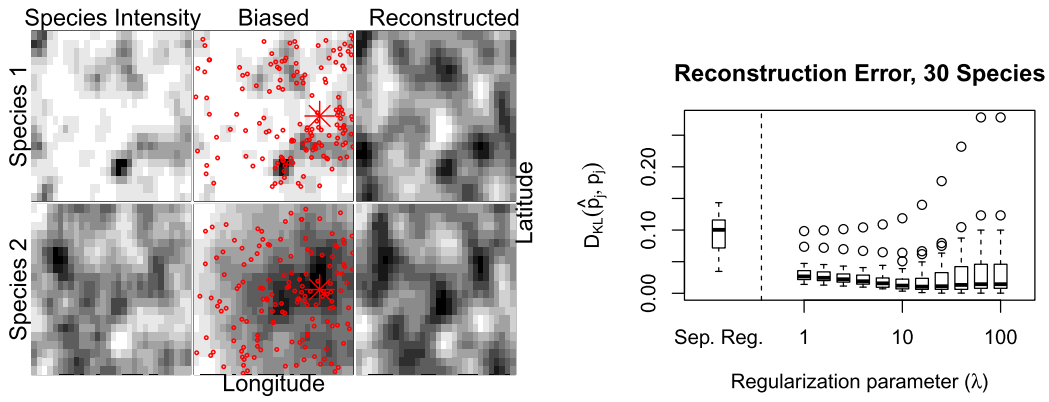
To borrow strength across species, we can model $B = UV'$, where $B = [\beta_1 \cdots \beta_m]$, and $U \in \mathbb{R}^{d\times r}$, $V \in \mathbb{R}^{m\times r}$. In effect we are positing that r latent habitat covariates $w(s) = x(s)'U$ can capture all of the important signal, with $V' = [v_1 \cdots v_m]$ representing the effect of the latent covariates on each species. Replacing the nonconvex rank constraint with a nuclear norm penalty, we arrive at a *reduced-rank Poisson regression* objective:

$$\min_{\alpha,\Theta,\delta} \mathcal{L}\big(\Pi_X\Theta + 1\zeta' + \delta' Z 1', Y\big) + \lambda\|\Theta\|_*, \tag{42}$$

where Π_X denotes projection onto the column space of X and $\zeta_j = \alpha_j + \epsilon_j$. Note that while this application is not posed directly as matrix completion as such, estimating $p_j(s)$ essentially amounts to predicting the locations of the missing observations.

We illustrate this method on simulation data, where the ground truth is known and estimation accuracy can be directly measured. On a 20×30 grid of pixels in the unit square $\mathcal{D} = [0,1] \times [0,1]$, we generate $d = 30$ covariates $x(s)$ as moving-average Gaussian processes. We then randomly generate latent factors U and species loadings V for each of $m = 30$ species, populating both matrices with i.i.d. Gaussian random variables. Next, we plant a "town" at location $s^* = (0.8, 0.5)$ and let $z(s) = -\|s - s^*\|_2^2$. Finally, we set α_j to normalize the intensities so that $\int_{\mathcal{D}} v_j(s)b_j(s) = 150$ for each species. Figure 2(a) shows the species intensity, biased intensity, and reconstructed species distribution for the first two species, for the best-performing value of λ on a validation set.

We compare our regularized estimator to a simpler method wherein we estimate β_j for $j = 1, \ldots, m$ via a separate log-linear Poisson regression for each species. To simplify estimation, the separate-regressions method is given perfect a priori knowledge of δ, so δ need not be estimated. Even with this advantage, the separate-regressions method overfits badly; 150 observations are not enough to accurately estimate a density with 30 parameters. Figure 2(b) shows a boxplot of the Kullback–Leibler distance $D_{\text{KL}}(\hat{p}_j \| p_j)$ for each value of λ, and for the separate regressions method. Figure 2(b) illustrates that combining multiple GLM models into a single RR-VGLM model can dramatically improve estimation performance, by borrowing strength across the different species. Note that unlike the phoneme data, these data are simulated so we know the assumptions of a low-rank structure are justified; we would not expect the RR-VGLM to improve on



(a) (b)

FIG. 2. *Results for ecological simulation.* (a) *Top row*: *The left panel shows* v_1, *the species intensity. The middle panel shows* $v_1 b_1$, *the biased intensity, with the observed process* $\mathcal{T}_1$ *plotted as red circles. The red star denotes the location of the "town" which drives the bias. The right panel shows the reconstructed species distribution* $\hat{p}_1$, *for the value of* λ *selected by cross-validation. Bottom row*: *Same quantities for species* 2. (b) *Comparison of the reconstruction error for the separate-regressions method versus the regularized method for a range of* λ *values. For each fit, the box summarizes* $D_{\mathrm{KL}}(\hat{p}_j, p_j)$ *for* $j = 1, \ldots, 30$. *The separate-regressions method performs quite poorly compared to the regularized method.*

separate GLMs if the species distributions shared no common structure.

6. DISCUSSION

We presented a framework for scalable convex optimization on matrix completion problems incorporating side information. The information can be diverse in its source, as long as it can be represented ultimately as some quadratic penalty which can be applied or inverted with ease. We have seen two examples where side information of different kinds is advantageous for predictive performance.

Although the bottleneck in our algorithm is an SVD of a large matrix, we can attain rapid convergence by exploiting the structure of the SVD target, which is often easy to apply.

6.1 When Is Side Information Helpful?

The phoneme data and ecological simulation provide two example problems where side information can be quite useful in providing better-targeted regularization for a specific scientific problem. Generally speaking, side information is likely to be helpful when we have good reason to believe that the row or column entities are related to each other in important ways that are likely to be reflected in the latent variables. In particular if rows or columns represent points in time or space, it will often be quite helpful to use our modeling framework to encode a smoothness assumption on the left or right singular vectors.

We can think of the advantages and disadvantages of including side information in terms of a typical bias-variance tradeoff. If we know ahead of time that several movies are in the same genre, or several genes are in the same pathway, then the methods in this article will let us borrow strength across the similar movies or genes to estimate their latent types. This borrowed strength is especially effective when some of the row or column entities are sparsely observed: for a sparsely observed row, its similarity to other, better observed rows may be the best information we have to go on in estimating its latent type. By contrast, if we have a great deal of information, then constraining latent row types to lie in a low-dimensional space can induce an estimation bias that hurts our predictive accuracy more than enough to counteract the variance reduction we achieve by borrowing strength.

In this paper, we have not discussed theoretical (statistical) properties of the estimators produced by the generalized nuclear norm regularized estimators. Understanding the theoretical statistical properties of the estimators proposed herein, formalizing the benefits of side-information is an interesting direction for future work.

6.2 Challenges and Potential Directions for Future Research

There are several important challenges to address before the framework described herein can see its full application in scientific problems. Statistically, we will often be interested in learning and interpreting the

latent factors, in addition to simply making predictions for held out data. To justify any interpretation of estimated factors, however, it is important to establish theoretical guarantees on consistency and estimation accuracy, as well as methods for uncertainty quantification, for example by extending recent advances in uncertainty quantification for PCA (Josse, Wager and Husson, 2016) and low-rank matrix completion (Carpentier et al., 2016). Finally, we are unaware of any methods for sensitivity analysis of the predictions or latent factors when the data may be MNAR; this may be an important topic for future work.

In addition, there are important avenues of computational research including systematic comparisons across the different competing methods described in Section 4.1 for the nuclear norm regularized problem, including matrix completion beyond the squared error loss function, especially when the gradient of the loss function is not Lipschitz. Developing specialized robust and scalable algorithms for the generalized nuclear norm problem is an interesting and in our opinion an important direction for future research—see Section 4.3 for specific details and outstanding challenges.

Temporal dynamics (Koren, 2010) play an important part in recommender systems and a key role in improving the predictive performance for the Netflix dataset (Bell and Koren, 2007, Koren, Bell and Volinsky, 2009)—it will be interesting to study if these models popularly used in the recommender systems community can be expressed in the generalized nuclear norm framework (or variations thereof) presented herein.

APPENDIX A: PROOF OF PROPOSITION 1

Let Π_P and Π_Q denote projections onto the images of P and Q. Then

$$\inf_{\Theta} \mathcal{L}(\Theta) + \|P\Theta Q\|_* \tag{43}$$

$$= \lim_{\varepsilon \downarrow 0} \inf_{\Theta} \mathcal{L}(\Theta) + \big\|(P + \varepsilon \Pi_P^{\perp})\Theta(Q + \varepsilon \Pi_Q^{\perp})\big\|_* \tag{44}$$

$$= \lim_{\varepsilon \downarrow 0} \inf_{U,V} \mathcal{L}(UV') + \frac{1}{2}\big\|(P + \varepsilon \Pi_P^{\perp})U\big\|_F^2 + \frac{1}{2}\big\|(Q + \varepsilon \Pi_Q^{\perp})V\big\|_F^2 \tag{45}$$

$$= \inf_{U,V} \mathcal{L}(UV') + \frac{1}{2}\|PU\|_F^2 + \frac{1}{2}\|QV\|_F^2. \tag{46}$$

We can see (45) by changing variables to $\widetilde{\Theta} = (P + \varepsilon \Pi_P^{\perp})^{-1}\Theta(Q + \varepsilon \Pi_Q^{\perp})^{-1}$, $\widetilde{U} = (P + \varepsilon \Pi_P^{\perp})^{-1}U$, and $\widetilde{V} = (Q + \varepsilon \Pi_Q^{\perp})^{-1}V$. (43) and (46) follow from the fact that $\|\Theta\|_* \geq \|\Pi\Theta\|_*$ for any projection Π (in this case Π_P), and similarly $\|U\|_F \geq \|\Pi U\|_F$.

For any Θ we can find $\Theta_1, \Theta_2, \Theta_3$ for which $\Theta = P^+\Theta_1 Q^+ + \Pi_P^{\perp}\Theta_2 + \Theta_3 \Pi_Q^{\perp}$. Then

$$\inf_{\Theta} \mathcal{L}(\Theta) + \|P\Theta Q\|_* \tag{47}$$

$$= \inf_{\Theta_1,\Theta_2,\Theta_3} \mathcal{L}\big(P^+\Theta_1 Q^+ + \Pi_P^{\perp}\Theta_2 + \Theta_3\Pi_Q^{\perp}\big) + \|\Pi_P\Theta_1\Pi_Q\|_* \tag{48}$$

$$= \inf_{\Theta_1,\Theta_2,\Theta_3} \mathcal{L}\big(P^+\Theta_1 Q^+ + \Pi_P^{\perp}\Theta_2 + \Theta_3\Pi_Q^{\perp}\big) + \|\Theta_1\|_*. \tag{49}$$

(47) holds because

$$P\big(P^+\Theta_1 Q^+ + \Pi_P^{\perp}\Theta_2 + \Theta_3\Pi_Q^{\perp}\big)Q = \Pi_P\Theta_1\Pi_Q.$$

(49) holds because $\|\Pi_P\Theta_1\Pi_Q\|_* \leq \|\Theta_1\|_*$ and we can attain the minimum by replacing Θ_1 with $\Pi_P\Theta_1\Pi_Q$, which does not change the $\mathcal{L}(\cdot)$ term.

APPENDIX B: SOLVING PROBLEM (8) WHEN P IS LOW RANK

Assuming wlog that P is low-rank with SVD $P = UDU'$ (with D a $J \times J$ diagonal matrix) the problem can be reformulated as:

$$\min_{\Gamma,\Theta} \mathcal{L}(\Theta; Y) + \lambda\|\Gamma\|_* \quad \text{s.t.} \quad DU'\Theta Q = \Gamma. \tag{50}$$

The above problem where Γ is a wide matrix with low rank (at most the rank of P) can be solved quite easily with a splitting method with the ADMM method (Boyd et al., 2011). A simple application of the ADMM procedure leads to Problem (50):

$$H(\Gamma, Z) = \mathcal{L}(\Theta; Y) + \lambda\|\Gamma\|_* + \langle Z, DU'\Theta Q - \Gamma\rangle + \frac{\rho}{2}\|DU'\Theta Q - \Gamma\|_F^2.$$

Note that Γ is a low rank rectangular matrix $\Gamma \in \mathbb{R}^{J\times m}$ with J small. The ADMM procedure requires optimizing $H(\Gamma, Z)$ w.r.t. Γ for Z fixed—this can be achieved easily using a proximal gradient method—the nuclear norm thresholding operation can be performed quite easily since it requires the SVD of a low-rank rectangular matrix (same dimension as Γ). The update step $\min_Z H(\Gamma, Z)$ with Γ held fixed can be achieved by solving a simple (unconstrained) convex function by performing gradient descent, for example.

We note that this problem can also be solved by using Proposition 1; and using proximal gradient descent methods on the reformulated problem of the form (11).

ACKNOWLEDGMENTS

A previous unpublished version of this manuscript is available on arxiv (Fithian and Mazumder, 2013). William Fithian was supported in part by NSF VIGRE grant DMS-0502385 and the Gerald J. Lieberman Fellowship. R. Mazumder is partially supported by ONR Grant N000141512342; and NSF IIS 1718258. We are grateful to Trevor Hastie and Julie Josse for helpful discussions, and to the reviewing team for their helpful feedback on an earlier draft.

REFERENCES

ABERNETHY, J., BACH, F., EVGENIOU, T. and VERT, J.-P. (2009). A new approach to collaborative filtering: Operator estimation with spectral regularization. *J. Mach. Learn. Res.* **10** 803–826.

AGGARWAL, C. C. and CHEN, B.-C. (2009). Regression-based latent factor models. In *Proceedings of the* 15*th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 19–28. ACM, New York.

AGARWAL, D. K. and CHEN, B.-C. (2015). *Statistical Methods for Recommender Systems*. Cambridge Univ. Press, Cambridge.

AGARWAL, D., ZHANG, L. and MAZUMDER, R. (2011). Modeling item–item similarities for personalized recommendations on Yahoo! front page. *Ann. Appl. Stat.* **5** 1839–1875. MR2884924

ANDERSON, T. W. (1951). Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Ann. Math. Stat.* **22** 327–351. MR0042664

ANGST, R., ZACH, C. and POLLEFEYS, M. (2011). The generalized trace-norm and its application to structure-from-motion problems. In 2011 *IEEE International Conference on Computer Vision* (*ICCV*) 2502–2509. IEEE, Los Alamitos, CA.

ATCHADÉ, Y. F., MAZUMDER, R. and CHEN, J. (2015). Scalable computation of regularized precision matrices via stochastic optimization. Preprint. Available at arXiv:1509.00426.

AUDIGIER, V., HUSSON, F. and JOSSE, J. (2016). A principal component method to impute missing values for mixed data. *Adv. Data Anal. Classif.* **10** 5–26. MR3464297

BECK, A. and TEBOULLE, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2** 183–202. MR2486527

BELL, R. M. and KOREN, Y. (2007). Lessons from the Netflix Prize Challenge. *ACM SIGKDD Explor. Newsl.* **9** 75–79.

BENNETT, J. and LANNING, S. (2007). The Netflix Prize. In *Proceedings of KDD Cup and Workshop* 3–6. ACM New York.

BERTSEKAS, D. P. (1999). *Nonlinear Programming*, 2nd ed. Athena Scientific, Belmont, MA. MR3444832

BERTSIMAS, D., COPENHAVER, M. S. and MAZUMDER, R. (2017). Certifiably optimal low rank factor analysis. *J. Mach. Learn. Res.* **18** Paper No. 29. MR3634896

BOTTOU, L. and BOUSQUET, O. (2008). The trade-offs of large scale learning. In *Advances in Neural Information Processing Systems* **20** (J. C. Platt, D. Koller, Y. Singer and S. T. Roweis, eds.) 161–168. MIT Press, Cambridge, MA.

BOYD, S. and VANDENBERGHE, L. (2004). *Convex Optimization*. Cambridge Univ. Press, Cambridge. MR2061575

BOYD, S., PARIKH, N., CHU, E., PELEATO, B. and ECKSTEIN, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3** 1–122.

BURER, S. and MONTEIRO, R. D. C. (2005). Local minima and convergence in low-rank semidefinite programming. *Math. Program.* **103** 427–444. MR2166543

CAI, T. and ZHOU, W.-X. (2013). A max-norm constrained minimization approach to 1-bit matrix completion. *J. Mach. Learn. Res.* **14** 3619–3647. MR3159403

CANDES, E. and PLAN, Y. (2010). Matrix completion with noise. *Proc. IEEE* **98** 925–936.

CANDÈS, E. J. and RECHT, B. (2009). Exact matrix completion via convex optimization. *Found. Comput. Math.* **9** 717–772. MR2565240

CANDÈS, E. J. and TAO, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Trans. Inform. Theory* **56** 2053–2080. MR2723472

CANDÈS, E. J., LI, X., MA, Y. and WRIGHT, J. (2011). Robust principal component analysis? *J. ACM* **58** Art. ID 11. MR2811000

CANDÈS, E. J., ELDAR, Y. C., STROHMER, T. and VORONINSKI, V. (2015). Phase retrieval via matrix completion [reprint of MR3032952]. *SIAM Rev.* **57** 225–251. MR3345342

CARPENTIER, A., KLOPP, O., LÖFFLER, M. and NICKL, R. (2016). Adaptive confidence sets for matrix completion. Preprint. Available at arXiv:1608.04861.

CHEN, Y. and WAINWRIGHT, M. J. (2015). Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. Preprint. Available at arXiv:1509.03025.

COTTET, V. and ALQUIER, P. (2018). 1-Bit matrix completion: PAC-Bayesian analysis of a variational approximation. *Mach. Learn.* **107** 579–603. MR3761297

DAVENPORT, M. A., PLAN, Y., VAN DEN BERG, E. and WOOTTERS, M. (2014). 1-bit matrix completion. *Inf. Inference* **3** 189–223. MR3311452

DE LEEUW, J. and VAN DER HEIJDEN, P. G. M. (1988). Correspondence analysis of incomplete contingency tables. *Psychometrika* **53** 223–233. MR0955467

DEVOLDER, O., GLINEUR, F. and NESTEROV, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Math. Program.* **146** 37–75. MR3232608

FAZEL, M. (2002). Matrix rank minimization with applications. Ph.D. thesis, Stanford Univ.

FITHIAN, W. and MAZUMDER, R. (2013). Scalable convex methods for flexible low-rank matrix modeling. Preprint. Available at arXiv:1308.4211.

FITHIAN, W., ELITH, J., HASTIE, T. and KEITH, D. A. (2015). Bias correction in species distribution models: Pooling survey and collection data for multiple species. *Methods Ecol. Evol.* **6** 424–438.

FOYGEL, R. and SREBRO, N. (2011). Concentration-based guarantees for low-rank matrix reconstruction. In *COLT* 315–340.

FRANK, M. and WOLFE, P. (1956). An algorithm for quadratic programming. *Nav. Res. Logist. Q.* **3** 95–110. MR0089102

FREUND, R. M., GRIGAS, P. and MAZUMDER, R. (2017). An extended Frank–Wolfe method with "in-face" directions, and its application to low-rank matrix completion. *SIAM J. Optim.* **27** 319–346. MR3615468

GERRISH, S. and BLEI, D. M. (2011). Predicting legislative roll calls from text. In *Proceedings of the* 28*th International Conference on Machine Learning* (*ICML*-11) 489–496.

GOLUB, G. H. and VAN LOAN, C. F. (1983). *Matrix Computations*. *Johns Hopkins Series in the Mathematical Sciences* **3**. Johns Hopkins Univ. Press, Baltimore, MD. MR0733103

HASTIE, T., BUJA, A. and TIBSHIRANI, R. (1995). Penalized discriminant analysis. *Ann. Statist.* **23** 73–102. MR1331657

HASTIE, T., TIBSHIRANI, R. and WAINWRIGHT, M. (2015). *Statistical Learning with Sparsity*: *The Lasso and Generalizations*. *Monographs on Statistics and Applied Probability* **143**. CRC Press, Boca Raton, FL. MR3616141

HASTIE, T., MAZUMDER, R., LEE, J. D. and ZADEH, R. (2015). Matrix completion and low-rank SVD via fast alternating least squares. *J. Mach. Learn. Res.* **16** 3367–3402. MR3450542

HUBER, P. J. (2011). *Robust Statistics*. Springer, New York.

JAGGI, M. and SULOVSK, M. (2010). A simple algorithm for nuclear norm regularized problems. In *Proceedings of the* 27*th International Conference on Machine Learning* (*ICML*-10) 471–478.

JAIN, P., NETRAPALLI, P. and SANGHAVI, S. (2013). Low-rank matrix completion using alternating minimization (extended abstract). In *STOC'*13*—Proceedings of the* 2013 *ACM Symposium on Theory of Computing* 665–674. ACM, New York. MR3210828

JOSSE, J. and HUSSON, F. (2012). Handling missing values in exploratory multivariate data analysis methods. *J. SFdS* **153** 79–99. MR3008600

JOSSE, J., WAGER, S. and HUSSON, F. (2016). Confidence areas for fixed-effects PCA. *J. Comput. Graph. Statist.* **25** 28–48. MR3474035

JOURNÉE, M., BACH, F., ABSIL, P.-A. and SEPULCHRE, R. (2010). Low-rank optimization on the cone of positive semidefinite matrices. *SIAM J. Optim.* **20** 2327–2351. MR2678395

KESHAVAN, R. H., MONTANARI, A. and OH, S. (2010). Matrix completion from a few entries. *IEEE Trans. Inform. Theory* **56** 2980–2998. MR2683452

KLOPP, O., LAFOND, J., MOULINES, É. and SALMON, J. (2015). Adaptive multinomial matrix completion. *Electron. J. Stat.* **9** 2950–2975. MR3439190

KOREN, Y. (2010). Collaborative filtering with temporal dynamics. *Commun. ACM* **53** 89–97.

KOREN, Y., BELL, R. and VOLINSKY, C. (2009). Matrix factorization techniques for recommender systems. *Computer* **42** 30–37.

LAFOND, J. (2015). Low rank matrix completion with exponential family noise. Preprint. Available at arXiv:1502.06919.

LARSEN, R. M. (2004). PROPACK—Software for large and sparse SVD calculations. Available at http://sun.stanford.edu/~rmunk/PROPACK.

LEE, J., RECHT, B., SREBRO, N., TROPP, J. and SALAKHUTDINOV, R. (2010). Practical large-scale optimization for max-norm regularization. In *Advances in Neural Information Processing Systems* 1297–1305.

LESIEUR, T., KRZAKALA, F. and ZDEBOROVÁ, L. (2015). MMSE of probabilistic low-rank matrix estimation: Universality with respect to the output channel. In 2015 53*rd Annual Allerton Conference on Communication*, *Control*, *and Computing* (*Allerton*) 680–687. IEEE, Los Alamitos, CA.

MARDIA, K. V., KENT, J. T. and BIBBY, J. M. (1979). *Multivariate Analysis*. Academic Press, London. MR0560319

MARTIN, A. D. and QUINN, K. M. (2002). Dynamic ideal point estimation via Markov chain Monte Carlo for the US Supreme Court, 1953–1999. *Polit. Anal.* **10** 134–153.

MAZUMDER, R., HASTIE, T. and TIBSHIRANI, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *J. Mach. Learn. Res.* **11** 2287–2322. MR2719857

MAZUMDER, R., RADCHENKO, P. and DEDIEU, A. (2017). Subset selection with shrinkage: Sparse linear modeling when the SNR is low. Preprint. Available at arXiv:1708.03288.

MENON, A. K. and ELKAN, C. (2010). A log-linear model with latent features for dyadic prediction. In 2010 *IEEE* 10*th International Conference on Data Mining* (*ICDM*) 364–373. IEEE, Los Alamitos, CA.

NEGAHBAN, S. and WAINWRIGHT, M. J. (2012). Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *J. Mach. Learn. Res.* **13** 1665–1697. MR2930649

NESTEROV, Y. (2004). *Introductory Lectures on Convex Optimization*: *A Basic Course*. *Applied Optimization* **87**. Kluwer Academic, Boston, MA. MR2142598

NESTEROV, YU. (2005). Smooth minimization of non-smooth functions. *Math. Program.* **103** 127–152. MR2166537

PARKER, J. T., SCHNITER, P. and CEVHER, V. (2014a). Bilinear generalized approximate message passing—Part I: Derivation. *IEEE Trans. Signal Process.* **62** 5839–5853. MR3281527

PARKER, J. T., SCHNITER, P. and CEVHER, V. (2014b). Bilinear generalized approximate message passing—Part II: Applications. *IEEE Trans. Signal Process.* **62** 5854–5867. MR3281528

REINSEL, G. C. and VELU, R. P. (1998). *Multivariate Reduced-Rank Regression*: *Theory and Applications*. *Lecture Notes in Statistics* **136**. Springer, New York. MR1719704

RENNIE, J. and SREBRO, N. (2005). Fast maximum margin matrix factorization for collaborative prediction. In *ICML*.

ROWEIS, S. (1998). EM algorithms for PCA and SPCA. In *Advances in Neural Information Processing Systems* **10** 626–632. MIT Press, Cambridge, MA.

RUBIN, D. B. (1976). Inference and missing data. *Biometrika* **63** 581–592. With comments by R. J. A. Little and a reply by the author. MR0455196

SALAKHUTDINOV, R. and MNIH, A. (2008a). Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *Proceedings of the* 25*th International Conference on Machine Learning* 880–887. ACM, New York.

SALAKHUTDINOV, R. and MNIH, A. (2008b). Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems* **20** (J. C. Platt, D. Koller, Y. Singer and S. T. Roweis, eds.) 1257–1264. MIT Press, Cambridge, MA.

SALAKHUTDINOV, R. and SREBRO, N. (2010). Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. Preprint. Available at arXiv:1002.2780.

SREBRO, N., RENNIE, J. and JAAKKOLA, T. (2005). Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems* **17** 1329–1336. MIT Press, Cambridge, MA.

SREBRO, N. and SHRAIBMAN, A. (2005). Rank, trace-norm and max-norm. In *Learning Theory*. *Lecture Notes in Computer Science* **3559** 545–560. Springer, Berlin. MR2203286

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B* **58** 267–288. MR1379242

TIPPING, M. E. and BISHOP, C. M. (1999). Probabilistic principal component analysis. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **61** 611–622. MR1707864

TODESCHINI, A., CARON, F. and CHAVENT, M. (2013). Probabilistic low-rank matrix completion with adaptive spectral regularization algorithms. In *Advances in Neural Information Processing Systems* 845–853.

UDELL, M., HORN, C., ZADEH, R. and BOYD, S. (2016). Generalized low rank models. *Found. Trends Mach. Learn.* **9** 1–118.

YANG, Y., MA, J. and OSHER, S. (2013). Seismic data reconstruction via matrix completion. *Inverse Probl. Imaging* **7** 1379–1392. MR3180685

YEE, T. W. and HASTIE, T. J. (2003). Reduced-rank vector generalized linear models. *Stat. Model.* **3** 15–41. MR1977163

YUAN, M., EKICI, A., LU, Z. and MONTEIRO, R. (2007). Dimension reduction and coefficient estimation in multivariate linear regression. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **69** 329–346. MR2323756