This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2020.2970306, IEEE Signal Processing Letters

Generalization error bounds for kernel matrix completion and extrapolation

Pere Giménez-Febrer, Alba Pagès-Zamora, and Georgios B. Giannakis

***Abstract*—Prior information can be incorporated in matrix completion to improve estimation accuracy and extrapolate the missing entries. Reproducing kernel Hilbert spaces provide tools to leverage the said prior information, and derive more reliable algorithms. This paper analyzes the generalization error of such approaches, and presents numerical tests confirming the theoretical results.**

I. Introduction

Matrix completion (MC) deals with the recovery of missing entries in a matrix – a task emerging in several applications such as image restoration [1], collaborative filtering [2] or positioning [3]. MC relies on the low rank of data matrices to enable reliable, even exact [4], recovery of the full unknown matrix. Exploiting this property, mainstream approaches to MC involve the minimization of the nuclear norm [5], [6] or a surrogate involving the data matrix factorization into a product of two low-rank matrices [7], [8].

One main assumption in the aforementioned approaches to MC is that the unknown matrix is incoherent, meaning the entries of its singular vectors are uniformly distributed, which implies that matrices with structured form are not allowed. Such structures may be induced by prior information embedded in, e.g., graphs [9], dictionaries [10], or heuristic assumptions [11]. Main approaches to MC leverage prior information with proper regularization [12]–[15], or by restricting the solution space [16]–[19]; most can be unified using a reproducing kernel Hilbert space (RKHS) framework [17], [18], which presents theoretical tools to exploit prior information.

When analyzing the performance of MC algorithms, several works, e.g. [2], [5], [16], [20], provide sample complexity bounds; that is, the evolution of the distance to the optimum across the number of samples and iterations. Other analyses are based on the generalization error (GE) [21]–[23], a metric that measures the difference between the loss function applied to a training dataset, and its expected value [24]. When the probability distribution of the data is unknown, the expected value is replaced by the average loss on a testing dataset [25]. Due to the potentially large matrix sizes and the small size of the training dataset, it is important that the estimated matrix exhibits low GE in order to prevent overfitting.

In [18], we introduced a novel Kronecker kernel matrix completion and extrapolation (KKMCEX) algorithm. This algorithm relies on kernel ridge regression with equal number of coefficients and observations, thus being attractive for imputing matrices with a minimal number of observations. The present paper deals with GE analysis in MC with prior information, and derives GE bounds based on the transductive Rademacher complexity [25]. Moreover, it presents numerical tests demonstrating that the GE of KKMCEX is less dependent on matrix size, thus making it more reliable when dealing with large matrices with a few observations.

The rest of the paper is organized as follows. Section II introduces the MC algorithms with and without prior information, while Section III presents their GE analyses. Then, Section IV describes the numerical tests, and Section V offers conclusions and possible extensions.

P. Giménez-Febrer and A. Pagès-Zamora are with the SPCOM Group, Universitat Politècnica de Catalunya-Barcelona Tech, Spain.
G. B. Giannakis is with the Dept. of ECE and Digital Technology Center, University of Minnesota, USA.
This work is supported by ERDF funds (TEC2013-41315-R and TEC2016-75067-C4-2), the Catalan Government (2017 SGR 578), and NSF grants (1500713, 1514056, 1711471 and 1509040).

II. MC with prior information

Consider a matrix $\boldsymbol{M} = \boldsymbol{F} + \boldsymbol{E}$, where $\boldsymbol{F} \in \mathbb{R}^{N\times L}$ denotes an unknown rank r matrix, and $\boldsymbol{E}$ is a noise matrix. We can only observe a subset of the entries in $\boldsymbol{M}$ whose indices are given by the sampling set $\mathcal{S}_m \subseteq \{1,\dots,N\}\times\{1,\dots,L\}$ of cardinality $m = |\mathcal{S}_m|$. Factorizing the unknown matrix as $\boldsymbol{F} = \boldsymbol{W}\boldsymbol{H}^T$, where $\boldsymbol{W} \in \mathbb{R}^{N\times p}$, $\boldsymbol{H} \in \mathbb{R}^{L\times p}$ and $p \geq r$, the unknown entries can be recovered by estimating

$$\{\hat{\boldsymbol{W}},\hat{\boldsymbol{H}}\} = \underset{\substack{\boldsymbol{W}\in\mathbb{R}^{N\times p}\\ \boldsymbol{H}\in\mathbb{R}^{L\times p}}}{\arg\min} \left|\left|P_{\mathcal{S}_m}(\boldsymbol{M}-\boldsymbol{W}\boldsymbol{H}^T)\right|\right|_{\mathrm{F}}^2 + \mu\left(||\boldsymbol{W}||_{\mathrm{F}}^2 + ||\boldsymbol{H}||_{\mathrm{F}}^2\right) \tag{1}$$

where $||\cdot||_F$ denotes the Frobenius norm, $P_{\mathcal{S}_m}(\cdot)$ sets to zero the entries with index $(i,j) \notin \mathcal{S}_m$ and leaves the rest unchanged, while μ is a regularization scalar. Hereafter we refer to (1) as the base MC formulation, which can also be written with the nuclear norm as a regularizer through the property $||\boldsymbol{F}||_* = \min_{\boldsymbol{F}=\boldsymbol{W}\boldsymbol{H}^T} \frac{1}{2}\left(||\boldsymbol{W}||_{\mathrm{F}}^2 + ||\boldsymbol{H}||_{\mathrm{F}}^2\right)$ [22].

While the base MC formulation makes no use of prior information, kernel MC (KMC) incorporates such knowledge by means of kernel functions that measure similarities between points in their input spaces. Let $\mathcal{X} := \{x_1,\dots,x_N\}$ and $\mathcal{Y} := \{y_1,\dots,y_L\}$ be spaces of entities with one-to-one correspondence with the rows and columns of $\boldsymbol{F}$, respectively. Given the input spaces $\mathcal{X}$ and $\mathcal{Y}$, KMC defines the pair of RKHSs $\mathcal{H}_w := \left\{w:\ w(x) = \sum_{n=1}^N b_n\kappa_w(x,x_n),\ b_n\in\mathbb{R}\right\}$ and $\mathcal{H}_h := \left\{h:\ h(y) = \sum_{l=1}^L c_l\kappa_h(y,y_l),\ c_l\in\mathbb{R}\right\}$, where $\kappa_w : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$ and $\kappa_h : \mathcal{Y}\times\mathcal{Y} \to \mathbb{R}$ are kernel functions. Then, KMC postulates that the columns of the factor matrices in (1) are functions in $\mathcal{H}_w$ and $\mathcal{H}_h$. Thus, we write $\boldsymbol{W} = \boldsymbol{K}_w\boldsymbol{B}$ and $\boldsymbol{H} = \boldsymbol{K}_h\boldsymbol{C}$, where $\boldsymbol{B}$ and $\boldsymbol{C}$ are coefficient matrices, while $\boldsymbol{K}_w \in \mathbb{R}^{N\times N}$ and $\boldsymbol{K}_h \in \mathbb{R}^{L\times L}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2020.2970306, IEEE Signal Processing Letters

are the kernel matrices with entries $(\boldsymbol{K}_w)_{i,j} = \kappa_w(x_i, x_j)$ and $(\boldsymbol{K}_h)_{i,j} = \kappa_h(y_i, y_j)$. The KMC formulations proposed in [14], [17], recover the factor matrices as

$$\{\hat{\boldsymbol{W}},\hat{\boldsymbol{H}}\} = \underset{\substack{\boldsymbol{W}\in\mathbb{R}^{N\times p}\\ \boldsymbol{H}\in\mathbb{R}^{L\times p}}}{\arg\min} \left|\left|P_{\mathcal{S}_m}(\boldsymbol{M}-\boldsymbol{W}\boldsymbol{H}^T)\right|\right|_{\mathrm{F}}^2 + \mu\big(\mathrm{Tr}(\boldsymbol{W}^T\boldsymbol{K}_w^{-1}\boldsymbol{W}) + \mathrm{Tr}(\boldsymbol{H}^T\boldsymbol{K}_h^{-1}\boldsymbol{H})\big). \quad (2)$$

The coefficient matrices are obtained as $\hat{\boldsymbol{B}} = \boldsymbol{K}_w^{-1}\hat{\boldsymbol{W}}$ and $\hat{\boldsymbol{C}} = \boldsymbol{K}_h^{-1}\hat{\boldsymbol{H}}$, although this step is usually omitted [14], [17].

Algorithms solving (1) and (2) rely on alternating minimization schemes that do not converge to the optimum in a finite number of iterations [26]. To overcome this limitation and obtain a closed-form solution, we introduced the KKMCEX method [18]. Associated with entries of $\boldsymbol{F}$, consider the two-dimensional $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ with $f(x_i, y_j) = \boldsymbol{F}_{i,j}$, and

$$\mathcal{H}_f := \left\{ f : f(x,y) = \sum_{n=1}^{N}\sum_{l=1}^{L} d_{n,l}\kappa_f((x,x_n),(y,y_l)), d_{n,l} \in \mathbb{R} \right\}.$$

Upon vectorizing $\boldsymbol{F}$, we obtain $\boldsymbol{f} = \mathrm{vec}(\boldsymbol{F}) = \boldsymbol{K}_f\boldsymbol{d}$, where $\boldsymbol{K}_f$ has entries κ_f and $\boldsymbol{d} := [d_{1,1}, \ldots, d_{N,1}, \ldots, d_{N,L}]^T$. Accordingly, the data matrix is vectorized as $\overline{\boldsymbol{m}} = \boldsymbol{S}\,\mathrm{vec}(\boldsymbol{M})$, where $\boldsymbol{S}$ is an $m \times NL$ binary sampling matrix with a single nonzero entry per row, and $\bar{\boldsymbol{e}} = \boldsymbol{S}\,\mathrm{vec}(\boldsymbol{E})$. With these definitions, the signal model for the observed entries becomes

$$\overline{\boldsymbol{m}} = \boldsymbol{S}\boldsymbol{f} + \bar{\boldsymbol{e}} = \boldsymbol{S}\boldsymbol{K}_f\boldsymbol{d} + \bar{\boldsymbol{e}}. \quad (3)$$

Recovery of the vectorized matrix is then performed using the kernel ridge regression estimate of $\boldsymbol{d}$ given by

$$\hat{\boldsymbol{d}} = \arg\min_{\boldsymbol{d}\in\mathbb{R}^{NL}} ||\overline{\boldsymbol{m}} - \boldsymbol{S}\boldsymbol{K}_f\boldsymbol{d}||_2^2 + \mu\boldsymbol{d}^T\boldsymbol{K}_f\boldsymbol{d}. \quad (4)$$

The closed-form solution to (4) satisfies $\hat{\boldsymbol{d}} = \boldsymbol{S}^T\hat{\bar{\boldsymbol{d}}}$, where

$$\hat{\bar{\boldsymbol{d}}} = (\boldsymbol{S}\boldsymbol{K}_f\boldsymbol{S}^T + \mu\boldsymbol{I})^{-1}\overline{\boldsymbol{m}}. \quad (5)$$

Since (5) only depends on the observations in $\mathcal{S}_m$, KKMCEX can be equivalently rewritten as

$$\hat{\bar{\boldsymbol{d}}} = \arg\min_{\bar{\boldsymbol{d}}\in\mathbb{R}^n} \left|\left|\overline{\boldsymbol{m}} - \bar{\boldsymbol{K}}_f\bar{\boldsymbol{d}}\right|\right|_2^2 + \mu\bar{\boldsymbol{d}}^T\bar{\boldsymbol{K}}_f\bar{\boldsymbol{d}} \quad (6)$$

where $\bar{\boldsymbol{K}}_f = \boldsymbol{S}\boldsymbol{K}_f\boldsymbol{S}^T$. Given κ_w and κ_h, hereafter we assume $\kappa_f((x,x_n),(y,y_l)) = \kappa_w(x,x_n)\kappa_h(y,y_l)$ as a kernel, which corresponds to a kernel matrix $\boldsymbol{K}_f = \boldsymbol{K}_h \otimes \boldsymbol{K}_w$. We refer the interested reader to [18] for a more detailed explanation of KKMCEX and its implementation.

While this work focuses on MC, the KMC formulation in (2) is similar to tensor completion [27], which has a regularization term per dimension. Similarly, KKMCEX can be formulated with tensors in mind by forming $\boldsymbol{K}_f$ as the Kronecker product of three or more matrices, and the analysis of the ensuing section carries over readily to tensors as well.

III. Generalization error in MC

In this section, we derive bounds for the GE of base MC in (1), KMC in (2) and KKMCEX in (4) algorithms. Consider rewriting MC in the general form

$$\hat{\boldsymbol{F}} = \underset{\boldsymbol{F}\in\mathcal{F}}{\arg\min}\, \frac{1}{m}\sum\nolimits_{(i,j)\in\mathcal{S}_m} l(\boldsymbol{M}_{i,j}, \boldsymbol{F}_{i,j}) \quad (7)$$

where $l : \mathbb{R}\times\mathbb{R} \to \mathbb{R}$ denotes the loss, and $\mathcal{F}$ is the hypothesis class. For instance, choosing the square loss and setting the class to the set of matrices with a nuclear norm smaller than a constant t results in the base MC formulation (1). In order to come up with distribution-free claims for MC, one may resort to the transductive GE analysis [25]. In this scenario, we are given $\mathcal{S}_n = \mathcal{S}_m \cup \mathcal{S}_u$ of n data comprising the union of the training set $\mathcal{S}_m$ and the testing set $\mathcal{S}_u$, where $|\mathcal{S}_u| = u$. These data are taken without repetition, and the objective is to minimize the loss on the testing set. Thus, the GE is the difference between the testing and training loss functions

$$\frac{1}{u}\sum\nolimits_{(i,j)\in\mathcal{S}_u} l(\boldsymbol{M}_{i,j}, \hat{\boldsymbol{F}}_{i,j}) - \frac{1}{m}\sum\nolimits_{(i,j)\in\mathcal{S}_m} l(\boldsymbol{M}_{i,j}, \hat{\boldsymbol{F}}_{i,j}). \quad (8)$$

By making this difference small, we ensure that $\hat{\boldsymbol{F}}$ has good generalization properties, meaning we expect to obtain a similar empirical loss on a different testing set of samples. Since MC algorithms find their solution among a class of matrices under different restrictions or hypotheses, we are interested in bounding (8) for any matrix in the solution space. Before we present such bounds, we need to introduce the notion of transductive Rademacher complexity (TRC) as follows.

Definition 1. ***Transductive Rademacher complexity*** *[25] Given a set $\mathcal{S}_n = \mathcal{S}_m \cup \mathcal{S}_u$ with $q := \frac{1}{u} + \frac{1}{m}$, the TRC of a matrix class $\mathcal{F}$ is*

$$R_n(\mathcal{F}) = q\mathbb{E}_\sigma\left\{\sup_{\boldsymbol{F}\in\mathcal{F}}\sum\nolimits_{(i,j)\in\mathcal{S}_n}\sigma_{i,j}\boldsymbol{F}_{i,j}\right\} \quad (9)$$

where $\sigma_{i,j}$ is a Rademacher random variable that takes values $[-1, 1]$ with probability 0.5. We may also write (9) in vectorized form as $R_n(\mathcal{F}) = q\mathbb{E}_\sigma\left\{\sup_{\boldsymbol{F}\in\mathcal{F}}\boldsymbol{\sigma}^T\mathrm{vec}(\boldsymbol{F})\right\}$, where $\boldsymbol{\sigma} = vec(\boldsymbol{\Sigma})$, and $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times L}$ has entries $\boldsymbol{\Sigma}_{i,j} = \sigma_{i,j}$ if $(i,j) \in \mathcal{S}_n$, and $\boldsymbol{\Sigma}_{i,j} = 0$ otherwise.

TRC measures the expected maximum correlation between any function in the class and the random vector $\boldsymbol{\sigma}$. Intuitively, the greater this correlation is, the larger is the chance of finding a solution in the hypothesis class that will fit any observation draw, that is, $\hat{\boldsymbol{F}}_{i,j} \simeq \boldsymbol{M}_{i,j} \forall\,(i,j) \in \mathcal{S}_n$. Although TRC measures the ability to fit both the testing and training data at once, a model for $\boldsymbol{F}$ is learnt using only the training data. While having a small loss across all entries in $\mathcal{S}_n$ is desirable, making it too small can lead to overfitting, and an increased error when predicting entries outside $\mathcal{S}_n$. Using the TRC, the GE is bounded as follows.

Theorem 1. *[25] Let $\mathcal{F}$ be a matrix hypothesis class. For a loss function l with Lipschitz constant γ, and any $\boldsymbol{F} \in \mathcal{F}$, it holds with probability $1-\delta$ that*

$$\frac{1}{u}\sum\nolimits_{(i,j)\in\mathcal{S}_u} l(\boldsymbol{M}_{i,j}, \boldsymbol{F}_{i,j}) - \frac{1}{m}\sum\nolimits_{(i,j)\in\mathcal{S}_m} l(\boldsymbol{M}_{i,j}, \boldsymbol{F}_{i,j}) \le R_n(l\circ\mathcal{F}) + 5.05q\sqrt{\min(m,u)} + \sqrt{2q\ln(1/\delta)}\,. \quad (10)$$

Thm. 1 asserts that in order to bound the GE, it only suffices to bound the TRC. Moreover, using the contraction property, which states that $R_n(l\circ\mathcal{F}) \le \frac{1}{\gamma}R_n(\mathcal{F})$ [25], we only need to calculate the TRC of $\mathcal{F}$. Given that the same loss function is used in MC, KMC and KKMCEX, in order to assess the GE upper bound of the three methods we will pursue the TRC for the hypothesis class of each algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2020.2970306, IEEE Signal Processing Letters

A. Generalization error for base MC

In the base MC formulation (1), the hypothesis class is $\mathcal{F}_{MC} := \{\boldsymbol{F} : ||\boldsymbol{F}||_* \leq t,\ t \in \mathbb{R}\}$, where the value of t is regulated by μ. As derived in [21], the TRC for this class of matrices is bounded as

$$R_n(\mathcal{F}_{MC}) \leq q\mathbb{E}_\sigma \left\{ \sup_{\boldsymbol{F}\in\mathcal{F}_{MC}} ||\boldsymbol{\Sigma}||_2 ||\boldsymbol{F}||_* \right\} \leq Gqt(\sqrt{N}+\sqrt{L}) \quad (11)$$

where G is a universal constant. Since $q = \frac{1}{m} + \frac{1}{u}$, the bound in (11) decays as $\mathcal{O}(\frac{1}{m} + \frac{1}{u}) \subseteq \mathcal{O}\left(1/\min(m,u)\right)$ for fixed t, N and L. However, the GE does not since the sum of the second and third terms on the right-hand side of (10) decays as $\mathcal{O}(1/\sqrt{\min(m,u)})$. Thus, the size of the training and testing datasets should be equal for the GE bound to diminish with the number of samples as $\mathcal{O}(1/\sqrt{m})$.

For non-fixed N and L, the TRC bound also scales with the matrix dimensions. Moreover, note that the nuclear norm is $\mathcal{O}(\sqrt{NL})$ since $||\boldsymbol{F}||_{\rm F} \leq ||\boldsymbol{F}||_* \leq \sqrt{r}\,||\boldsymbol{F}||_{\rm F}$. Therefore, t should also grow with N and L in order to match the hypothesis class, and obtain a good estimate of $\boldsymbol{F}$. Hence, for varying N, L and n with $m = u$, the GE is $\mathcal{O}(\frac{1}{\sqrt{m}} + \frac{N\sqrt{L}+L\sqrt{N}}{m})$. This implies that increasing N or L results in a larger GE bound regardless of the value of n, whereas increasing m and u by the same amount results in a smaller GE bound.

B. Generalization error for KMC

Unlike base MC that minimizes the nuclear norm of the data matrix, KMC does not directly employ the rank in its objective function. Instead, it imposes constraints on the maximum norm of the factor matrices in their respective RKHSs. Hence, the TRC for KMC is bounded as follows.

Theorem 2. *If the KMC hypothesis class is* $\mathcal{F}_K := \{\boldsymbol{F} : \boldsymbol{F} = \boldsymbol{K}_w\boldsymbol{B}\boldsymbol{C}^T\boldsymbol{K}_h, \text{Tr}(\boldsymbol{B}^T\boldsymbol{K}_w\boldsymbol{B}) + \text{Tr}(\boldsymbol{C}^T\boldsymbol{K}_h\boldsymbol{C}) < t_B\}$*, then*

$$R_n(\mathcal{F}_K) \leq \lambda_{\max} Gqt_B(\sqrt{N}+\sqrt{L}) \quad (12)$$

where $\lambda_{\max}$ *is the largest eigenvalue of* $\boldsymbol{K}_w$ *and* $\boldsymbol{K}_h$*.*

Proof. Rewrite the nuclear norm in (11) as

$$||\boldsymbol{F}||_* = \frac{1}{2}(||\boldsymbol{W}||_{\rm F}^2 + ||\boldsymbol{H}||_{\rm F}^2) = \frac{1}{2}(\text{Tr}(\boldsymbol{B}^T\boldsymbol{K}_w^2\boldsymbol{B}) + \text{Tr}(\boldsymbol{C}^T\boldsymbol{K}_h^2\boldsymbol{C}))$$
$$\leq \frac{\lambda_{\max}}{2}[\text{Tr}(\boldsymbol{B}^T\boldsymbol{K}_w\boldsymbol{B}) + \text{Tr}(\boldsymbol{C}^T\boldsymbol{K}_h\boldsymbol{C})] \leq \frac{\lambda_{\max}t_B}{2} \quad (13)$$

where we used that $\text{Tr}(\boldsymbol{B}^T\boldsymbol{K}_w^2\boldsymbol{B}) = \sum_{i=1}^N \boldsymbol{b}_i^T\boldsymbol{K}_w^2\boldsymbol{b}_i$ with $\boldsymbol{b}_i$ denoting the i^{th} column of $\boldsymbol{B}$, and $\boldsymbol{b}_i^T\boldsymbol{K}_w^{\frac{1}{2}}\boldsymbol{K}_w\boldsymbol{K}_w^{\frac{1}{2}}\boldsymbol{b}_i \leq \lambda_{\max}\boldsymbol{b}_i^T\boldsymbol{K}_w\boldsymbol{b}_i$. □

Thm. 2 establishes that the TRC bound expressions of KMC and base MC are identical within a scale. With $t_B = t$, $\lambda_{\max}$ controls whether KMC has a larger or smaller TRC bound than base MC. Thus, according to Thm. 2, the GE bound for KMC shrinks with n and grows with N, L and $\lambda_{\max}$. Next, we derive an alternative bound in order to gain further insights about the factors affecting the GE.

Consider the factorizations $\boldsymbol{K}_w = \boldsymbol{\Phi}_w\boldsymbol{\Phi}_w^T$ and $\boldsymbol{K}_h = \boldsymbol{\Phi}_h\boldsymbol{\Phi}_h^T$, where $\boldsymbol{\Phi}_w \in \mathbb{R}^{N\times d_w}$ and $\boldsymbol{\Phi}_h \in \mathbb{R}^{L\times d_h}$. Plugging these into (2) and setting $\boldsymbol{W} = \boldsymbol{K}_w\boldsymbol{B}$ and $\boldsymbol{H} = \boldsymbol{K}_h\boldsymbol{C}$, yields

$$\left|\left|P_{\mathcal{S}_m}(\boldsymbol{M} - \boldsymbol{\Phi}_w\boldsymbol{\Phi}_w^T\boldsymbol{B}\boldsymbol{C}^T\boldsymbol{\Phi}_h\boldsymbol{\Phi}_h^T)\right|\right|_{\rm F}^2 + \mu\big(\text{Tr}(\boldsymbol{B}^T\boldsymbol{\Phi}_w\boldsymbol{\Phi}_w^T\boldsymbol{B}) + \text{Tr}(\boldsymbol{C}^T\boldsymbol{\Phi}_h\boldsymbol{\Phi}_h^T\boldsymbol{C})\big) \quad (14)$$
$$= \left|\left|P_{\mathcal{S}_m}(\boldsymbol{M} - \boldsymbol{\Phi}_w\boldsymbol{A}_w\boldsymbol{A}_h^T\boldsymbol{\Phi}_h^T)\right|\right|_{\rm F}^2 + \mu\Big(||\boldsymbol{A}_w||_{\rm F}^2 + ||\boldsymbol{A}_h||_{\rm F}^2\Big) \quad (15)$$

where $\boldsymbol{A}_w = \boldsymbol{\Phi}_w^T\boldsymbol{B}$ and $\boldsymbol{A}_h = \boldsymbol{\Phi}_h^T\boldsymbol{C}$ are coefficient matrices of size $d_w \times p$ and $d_h \times p$, respectively. Optimizing for $\{\boldsymbol{B}, \boldsymbol{C}\}$ in (14) or for $\{\boldsymbol{A}_w, \boldsymbol{A}_h\}$ in (15) yields the same $\hat{\boldsymbol{F}}$ provided that $\{\boldsymbol{\Phi}_w^T, \boldsymbol{\Phi}_h^T\}$ have full column rank. Under this assumption, we consider the hypothesis class $\mathcal{F}_I := \Big\{\boldsymbol{F} : \boldsymbol{F} = \boldsymbol{\Phi}_w\boldsymbol{A}_w\boldsymbol{A}_h^T\boldsymbol{\Phi}_h^T, ||\boldsymbol{A}_w||_{\rm F}^2 \leq t_w, ||\boldsymbol{A}_h||_{\rm F}^2 < t_h\Big\}$, which satisfies $\mathcal{F}_I = \mathcal{F}_K$. Clearly, (15) is the objective used by the inductive MC [16] method; and therefore, we have shown that inductive MC is a special case of KMC. This leads to the following result.

Theorem 3. *If* $\boldsymbol{K} = (\boldsymbol{\Phi}_h \otimes \boldsymbol{\Phi}_w)(\boldsymbol{\Phi}_h \otimes \boldsymbol{\Phi}_w)^T$*, and* $\boldsymbol{S}_n$ *is a binary sampling matrix that selects the entries in* $\mathcal{S}_n$*, then*

$$R_n(\mathcal{F}_I) \leq q\sqrt{t_w t_h}\text{Tr}(\sqrt{\boldsymbol{S}_n\boldsymbol{K}\boldsymbol{S}_n^T}). \quad (16)$$

Proof. With $\boldsymbol{\sigma} := \text{vec}(\boldsymbol{\Sigma})$, $b_w := ||\boldsymbol{A}_w||_{\rm F}^2$, and $b_h := ||\boldsymbol{A}_h||_{\rm F}^2$, we have that

$$R_n(\mathcal{F}_I) = q\mathbb{E}_\sigma \left\{ \sup_{b_w\leq t_w, b_h\leq t_h} \boldsymbol{\sigma}^T\text{vec}(\boldsymbol{\Phi}_w\boldsymbol{A}_w\boldsymbol{A}_h^T\boldsymbol{\Phi}_h^T) \right\}$$
$$= q\mathbb{E}_\sigma \left\{ \sup_{b_w\leq t_w, b_h\leq t_h} \boldsymbol{\sigma}^T(\boldsymbol{\Phi}_h \otimes \boldsymbol{\Phi}_w)\text{vec}(\boldsymbol{A}_w\boldsymbol{A}_h^T) \right\}$$
$$\leq q\mathbb{E}_\sigma \left\{ \sup_{b_w\leq t_w, b_h\leq t_h} \left|\left|\boldsymbol{\sigma}^T(\boldsymbol{\Phi}_h \otimes \boldsymbol{\Phi}_w)\right|\right|_2 \left|\left|\text{vec}(\boldsymbol{A}_w\boldsymbol{A}_h^T)\right|\right|_2 \right\}$$
$$= q\mathbb{E}_\sigma \left\{ \sup_{b_w\leq t_w, b_h\leq t_h} \sqrt{\boldsymbol{\sigma}^T\boldsymbol{K}\boldsymbol{\sigma}}\left|\left|\boldsymbol{A}_w\boldsymbol{A}_h^T\right|\right|_{\rm F} \right\}$$
$$\leq q\mathbb{E}_\sigma \left\{ \sup_{b_w\leq t_w, b_h\leq t_h} \sqrt{\boldsymbol{\sigma}^T\boldsymbol{K}\boldsymbol{\sigma}}\,||\boldsymbol{A}_w||_{\rm F}\left|\left|\boldsymbol{A}_h^T\right|\right|_{\rm F} \right\}$$
$$\leq q\sqrt{t_w t_h}\sqrt{\mathbb{E}_\sigma\{\boldsymbol{\sigma}^T\boldsymbol{K}\boldsymbol{\sigma}\}} = q\sqrt{t_w t_h}\sqrt{\text{Tr}(\boldsymbol{S}_n\boldsymbol{K}\boldsymbol{S}_n^T)}$$

where we have used the Cauchy-Schwarz inequality, the sub-multiplicative property of the Frobenius norm, and Jensen's inequality in the first, second and third inequalities. □

Theorem 3 shows through $\boldsymbol{S}_n$ how the choice of sampling and testing datasets impacts the TRC bound, which can be leveraged to develop optimal sampling strategies [28]. Moreover, it reveals the conditions under which the GE bound does not grow with N and L, which are as follows.

If c denotes the maximum value of the sampled entries in the diagonal of $\boldsymbol{K}$, and $m = u$, then Theorem 3 provides a bound that decays as $\mathcal{O}(\sqrt{\frac{t_w t_h c}{m}})$. The definition of $\mathcal{F}_I$ implies that t_w and t_h are determined by the Frobenius norms of $\{\boldsymbol{A}_w, \boldsymbol{A}_h\}$. Since $||\boldsymbol{A}_w||_F^2$ and $||\boldsymbol{A}_h||_F^2$ are $\mathcal{O}(d_w p)$ and $\mathcal{O}(d_h p)$, respectively, the TRC bound is limited by the rank of the kernel matrices, given by d_w and d_h. Therefore, the GE bound for KMC in (10) scales as $\mathcal{O}(\sqrt{\frac{d_w d_h p^2 c}{m}})$, which is maintained through different N and L so long as the kernel matrices have constant rank, and c does not change.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2020.2970306, IEEE Signal Processing Letters

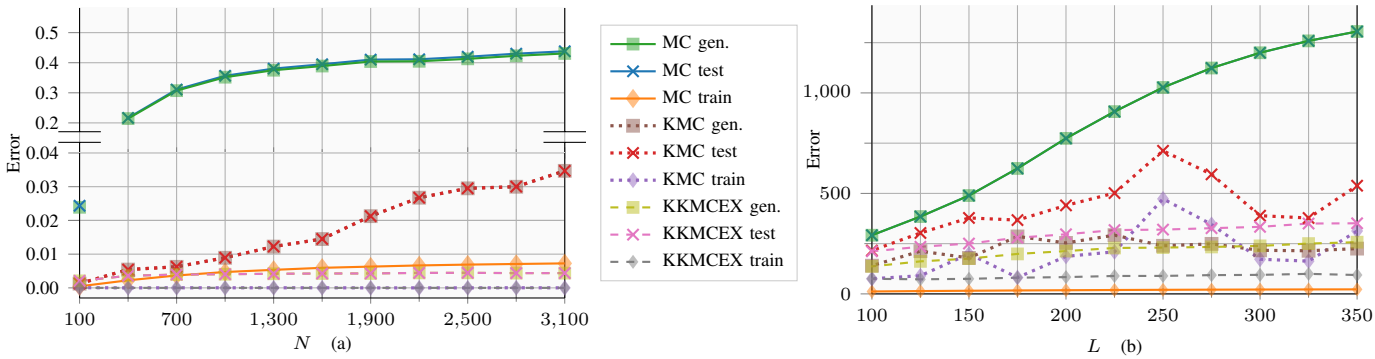

Figure 1: Training loss, testing loss, and generalization error vs. matrix size for: (a) synthetic data, and (b) temperature data.

C. Generalization error for KKMCEX

Although KMC and KKMCEX provide an estimate within the same RKHS since $\boldsymbol{K}_f = \boldsymbol{K}_h \otimes \boldsymbol{K}_w$, the complexity of the hypothesis spaces differs. This results in a TRC bound for KKMCEX that is given by the theorem next.

Theorem 4. *If* $\mathcal{F}_R := \{\boldsymbol{F} : \boldsymbol{F} = \text{unvec}(\boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{d}}), \bar{\boldsymbol{d}}^T \bar{\boldsymbol{K}}_f \bar{\boldsymbol{d}} \leq b^2,\ b \in \mathbb{R}\}$ *is the hypothesis class for KKMCEX, it holds that*

$$R_n(\mathcal{F}_R) \leq qb\sqrt{\text{Tr}(\boldsymbol{S}_n \boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{K}}_f^{-1} \boldsymbol{S} \boldsymbol{K}_f \boldsymbol{S}_n^T)}. \tag{17}$$

Proof.

$$\begin{aligned} R_n(\mathcal{F}_R) &= q\mathbb{E}_\sigma \left\{ \sup_{\bar{\boldsymbol{d}}^T \boldsymbol{K}_f \bar{\boldsymbol{d}} \leq b} \boldsymbol{\sigma}^T \boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{d}} \right\} \\ &= q\mathbb{E}_\sigma \left\{ \sup_{\bar{\boldsymbol{d}}^T \bar{\boldsymbol{K}}_f \bar{\boldsymbol{d}} \leq b} \boldsymbol{\sigma}^T \boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{K}}_f^{-\frac{1}{2}} \bar{\boldsymbol{K}}_f^{\frac{1}{2}} \bar{\boldsymbol{d}} \right\} \\ &\leq q\mathbb{E}_\sigma \left\{ \sup_{\bar{\boldsymbol{d}}^T \bar{\boldsymbol{K}}_f \bar{\boldsymbol{d}} \leq b} \left\| \boldsymbol{\sigma}^T \boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{K}}_f^{-\frac{1}{2}} \right\|_2 \left\| \bar{\boldsymbol{K}}_f^{\frac{1}{2}} \bar{\boldsymbol{d}} \right\|_2 \right\} \\ &\leq qb\mathbb{E}_\sigma \left\{ \left\| \boldsymbol{\sigma}^T \boldsymbol{K}_f \boldsymbol{S}^T \bar{\boldsymbol{K}}_f^{-\frac{1}{2}} \right\|_2 \right\} \\ &= qb\sqrt{\text{Tr}(\boldsymbol{S}_n \boldsymbol{K}_f \boldsymbol{S} \bar{\boldsymbol{K}}_f^{-1} \boldsymbol{S}^T \boldsymbol{K}_f \boldsymbol{S}_n^T)}. \end{aligned} \tag{18}$$

□

Supposing that the entries of $\boldsymbol{K}_f$ have maximum value c, the bound in (17) decays as $\mathcal{O}(\sqrt{nc}/\min(m,u))$. For $m = u$, this yields a rate $\mathcal{O}(\sqrt{\frac{c}{m}})$. Thus, the GE bound induced by (17) only scales with the number of samples provided that c is constant for different N and L. Interestingly, although the degrees of freedom of KKMCEX (and hence the risk of overfitting) grow with m, the GE does not increase because the number of samples increases proportionally. Thus, different from baseline MC and KMC, similar performance is expected on the testing dataset regardless of the data matrix size.

IV. Numerical tests

This section compares the GE of base MC and KMC, solved via alternating least-squares (ALS) [26], with the KKMCEX solved with (5). Besides comparing the GE of these algorithms, we also assess how the matrix size impacts the GE. To this end, we first use a fixed-rank synthetic data matrix with $N = L$ generated as $\boldsymbol{F} = \boldsymbol{K}_w \boldsymbol{B}\boldsymbol{C}^T \boldsymbol{K}_h$. The kernel matrices are $\boldsymbol{K}_w = \boldsymbol{K}_h = \text{abs}(\boldsymbol{R}\boldsymbol{D}\boldsymbol{R}^T)$, where $\boldsymbol{R} \in \mathbb{C}^{N \times N}$ is the DFT basis and $\boldsymbol{D} \in \mathbb{R}^{N \times N}$ is a diagonal matrix with decreasing values on its diagonal. The coefficient matrices $\{\boldsymbol{B}, \boldsymbol{C}\}$ have $p = 30$ columns, with entries drawn from a zero-mean Gaussian distribution with variance 1. The tests are run over 1,000 realizations. A new matrix $\boldsymbol{F}$ is generated per realization with $m = 1,000$ entries drawn uniformly at random, and the remaining $u = N^2 - m$ forming the testing dataset. The parameter μ is chosen by cross-validation for each size.

Fig. 1a shows the training, testing, and GEs for the synthetic matrices. We observe for base MC that the training loss is small, whereas it is much larger on the testing dataset, and also it grows with N. Moreover, since the training loss is minimal, the GE coincides with the testing loss. Clearly, the base MC solution (1) is not able to predict the unobserved entries due to the lack of prior information that would allow for extrapolation. In addition, the GE approaches saturation for large matrix sizes since most entries in the estimated matrix are 0, and the testing loss tends to the average $\frac{1}{u}\sum_{(i,j)\in\mathcal{S}_u} \boldsymbol{M}_{i,j}^2$. Regarding KMC and KKMCEX, we observe that both algorithms achieve a constant training loss. Although not visible on the plot, the training loss of KKMCEX is one order of magnitude smaller than that of KMC. On the other hand, the testing and GE of KKMCEX are constant unlike in KMC for which both are higher and grow with N. These results confirm what was asserted by the GE bounds in Section III.

Fig. 1b shows the numerical tests with an $150 \times L$ matrix of temperature measurements [18] taken in 2002 by 150 weather stations in the US. The kernel matrices are the row and column covariances of the same data from 2001, and $m = 500$ with $u = 150L - m$. We observe that for KMC and KKMCEX both training and testing errors grow with L. However, KMC is unstable both in training and testing, whereas KKMCEX is smooth. Thus, although the GE grows slightly for KKMCEX, it is more reliable when the number of samples is very small.

V. Conclusions

This work analyzed the GE for MC with prior information following a procedure that can be utilized to additional data imputation methods after properly defining a loss function and corresponding hypothesis class. Bounds on the TRC have established that baseline MC and KMC become less reliable as the size of the matrix increases when the number of samples remains constant. On the other hand, KKMCEX offers improved analytical guarantees with a GE that scales only with the number of samples. Moreover, numerical tests have corroborated the theoretical findings for synthetic data with known kernel matrices, and have demonstrated improved performance for KKMCEX with real data. Finally, since the RKHS framework generalizes several MC settings with prior information, the analysis herein applies also to these settings.

REFERENCES

[1] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. of Computer Vision and Pattern Recognition Conf.*, San Francisco, USA, Jun. 2010, pp. 1791–1798.

[2] N. Rao, H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon, "Collaborative filtering with graph information: Consistency and scalable methods," in *Advances in Neural Information Processing Systems*, Montreal, Canada, Dec. 2015, pp. 2107–2115.

[3] T. L. Nguyen and Y. Shin, "Matrix completion optimization for localization in wireless sensor networks for intelligent IoT," *Sensors (Switzerland)*, vol. 16, no. 5, pp. 1–11, 2016.

[4] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, Dec. 2009.

[5] J. F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, Jan. 2010.

[6] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, no. 1-2, pp. 321–353, Jun. 2011.

[7] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[8] R. Sun, "Matrix Completion via Nonconvex Factorization: Algorithms and Theory," Ph.D. dissertation, UNIVERSITY OF MINNESOTA, 2015.

[9] V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Matrix completion on graphs," in *Neural Information Processing Systems Workshop "Out of the Box: Robustness in High Dimension"*, Montreal, Canada, Dec. 2014.

[10] K. Yi, J. Wan, T. Bao, and L. Yao, "A DCT regularized matrix completion algorithm for energy efficient data gathering in wireless sensor networks," *Int. Journal of Distributed Sensor Networks*, vol. 11, no. 7, p. 272761, Jul. 2015.

[11] J. Cheng, Q. Ye, H. Jiang, D. Wang, and C. Wang, "STCDG: an efficient data gathering algorithm based on matrix completion for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 850–861, Feb. 2013.

[12] S. Chen, A. Sandryhaila, J. M. Moura, and J. Kovacević, "Signal recovery on graphs: Variation minimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 17, pp. 4609–4624, Sep. 2015.

[13] A. Gogna and A. Majumdar, "Matrix completion incorporating auxiliary information for recommender system design," *Expert Systems with Applications*, vol. 42, no. 14, pp. 5789–5799, Aug. 2015.

[14] T. Zhou, H. Shan, A. Banerjee, and G. Sapiro, "Kernelized probabilistic matrix factorization: Exploiting graphs and side information," in *Proc. of SIAM Int. Conf. on Data Mining*, Minneapolis, USA, Jul. 2012, pp. 403–414.

[15] P. Giménez-Febrer and A. Pagès-Zamora, "Matrix completion of noisy graph signals via proximal gradient minimization," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, New Orleans, USA, March 2017, pp. 4441–4445.

[16] P. Jain and I. S. Dhillon, "Provable inductive matrix completion," *arXiv preprint arXiv:1306.0626*, 2013.

[17] J. A. Bazerque and G. B. Giannakis, "Nonparametric basis pursuit via sparse kernel-based learning: A unifying view with advances in blind methods," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 112–125, Jul. 2013.

[18] P. Giménez-Febrer, A. Pagès-Zamora, and G. B. Giannakis, "Matrix completion and extrapolation via kernel regression," *IEEE Transactions on Signal Processing*, vol. 67, no. 19, pp. 5004–5017, Oct 2019.

[19] J. Abernethy, F. Bach, T. Evgeniou, and J.-P. Vert, "Low-rank matrix factorization with attributes," *arXiv preprint cs/0611124*, 2006.

[20] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proc. of the IEEE*, vol. 98, no. 6, pp. 925–936, Jun. 2010.

[21] O. Shamir and S. Shalev-Shwartz, "Matrix Completion with the Trace Norm: Learning, Bounding, and Transducing," *Journal of Machine Learning Research*, vol. 15, pp. 3401–3423, 2014.

[22] N. Srebro and A. Shraibman, "Rank, Trace-Norm and Max-Norm," Tech. Rep.

[23] R. Foygel and N. Srebro, "Concentration-based guarantees for low-rank matrix reconstruction," in *Proc. of the 24th Annual Conf. on Learning Theory*, 2011, pp. 315–340.

[24] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004, ch. 4.

[25] R. El-Yaniv and D. Pechyony, "Transductive Rademacher complexity and its applications," *Journal of Artificial Intelligence Research*, vol. 35, pp. 193–234, 2009.

[26] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proc. of ACM Symp. on Theory of Computing*, Palo Alto, USA, Jun. 2013, pp. 665–674.

[27] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Rank regularization and bayesian inference for tensor completion and extrapolation," *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5689–5703, Nov 2013.

[28] Q. Gu and J. Han, "Towards active learning on graphs: An error bound minimization approach," in *2012 IEEE 12th International Conference on Data Mining*. IEEE, 2012, pp. 882–887.