From Micro to Macro: Data Driven Phenotyping by Densification of Longitudinal Electronic Medical Records

Jiayu Zhou[1,2], Fei Wang[3], Jianying Hu[3], Jieping Ye[1,2]
[1]Center for Evolutionary Medicine and Informatics, The Biodesign Institute, ASU, Tempe, AZ
[2]Department of Computer Science and Engineering, ASU, Tempe, AZ
[3]Healthcare Analytics, IBM T.J. Watson Research Center, Yorktown Heights, NY

ABSTRACT

Inferring phenotypic patterns from population-scale clinical data is a core computational task in the development of personalized medicine. One important source of data on which to conduct this type of research is patient Electronic Medical Records (EMR). However, the patient EMRs are typically sparse and noisy, which creates significant challenges if we use them directly to represent patient phenotypes. In this paper, we propose a data driven phenotyping framework called PACIFIER (PAtient reCord densIFIER), where we interpret the longitudinal EMR data of each patient as a sparse matrix with a feature dimension and a time dimension, and derive more robust patient phenotypes by exploring the latent structure of those matrices. Specifically, we assume that each derived phenotype is composed of a subset of the medical features contained in original patient EMR, whose value evolves smoothly over time. We propose two formulations to achieve such goal. One is Individual Basis Approach (IBA), which assumes the phenotypes are different for every patient. The other is Shared Basis Approach (SBA), which assumes the patient population shares a common set of phenotypes. We develop an efficient optimization algorithm that is capable of resolving both problems efficiently. Finally we validate PACIFIER on two real world EMR cohorts for the tasks of early prediction of Congestive Heart Failure (CHF) and End Stage Renal Disease (ESRD). Our results show that the predictive performance in both tasks can be improved significantly by the proposed algorithms (average AUC score improved from 0.689 to 0.816 on CHF, and from 0.756 to 0.838 on ESRD respectively, on diagnosis group granularity). We also illustrate some interesting phenotypes derived from our data.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*; J.3 [**Life and Medical Sciences**]: Health, Medical information systems

KDD'14, August 24–27, 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2623330.2623711.

Keywords

Medical informatics; phenotyping; sparse learning; matrix completion; densification

1. INTRODUCTION

Patient *Electronic Medical Records* (EMR) are systematic collections of longitudinal patient health information generated from one or more encounters in any care delivery setting. Typical information contained in EMR includes patient demographics, encounter records, progress notes, problems, medications, vital signs, immunizations, laboratory data and radiology reports, and etc. Effective utilization of EMR is the key to many medical informatics research problems, such as predictive modeling [36], disease early detection [30], comparative effectiveness research [17] and risk stratification [19].

Working directly with raw EMR is very challenging because it is usually sparse, noisy and irregular. Deriving better and more robust representation of the patients, or phenotyping, is very important in many medical informatics applications [13, 35]. One significant challenge for phenotyping with longitudinal EMR is *data sparsity*. To illustrate this, we show the EMR of a Congestive Heart Failure (CHF) patient in Fig.1, which is represented as a matrix. The horizontal axis is time with the granularity of days. The vertical axis is a set of medical events, which in this example is a set of diagnosis codes. Each dot in a matrix indicates that the corresponding diagnosis is observed for this patient at the corresponding day. From the figure we can see that there are only 37 nonzero entries within a 90-day window.

With those sparse matrices, many existing works just treat those zero values as actual zeros [30, 27, 25], and construct feature vectors from them with some summary statistics, then feed those feature vectors into computational models (e.g., classification, regression and clustering) for specific tasks. However, this may not be appropriate because many of those zero entries are not actual zeros but missing (the patient did not pay a visit and thus there is no corresponding record). Thus, the feature vectors constructed in this way are not accurate. As a consequence, the performance of the computational models will be compromised.

To handle the sparsity problem, we propose a general framework, PACIFIER (PAtient reCord densIFIER), for phenotyping patients with their EMRs, which imputes the values of those missing entries by exploring the latent structures on both feature and time dimensions. Specifically, we assume those observed medical features in EMR (microphenotypes) can be mapped to some latent medical con-

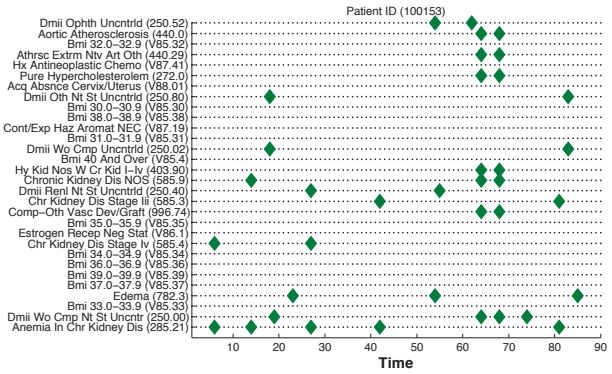


Figure 1: An example of the patient's EMR. The horizontal axis represents the number of days since the patient has records. The vertical axis corresponds to different diagnosis codes. A green diamond indicates the corresponding code is diagnosed for this patient at the corresponding day.

cept space with a much lower dimensionality, such that each medical concept can be viewed as a combination of several observed medical features (macro-phenotypes). In this way, we expect to discover a much denser representation of the patient EMR in the latent space, and the values of those medical concepts evolve smoothly over time. We develop the following two specific formulations to achieve such goal:

- *Individual Basis Approach* (IBA), which approximates each individual EMR matrix as the product of two latent matrices. One is the mapping from those observed medical features to the latent medical concepts, the other describes how the values of those medical concepts evolve over time.
- *Shared Basis Approach* (SBA), which also approximates the EMR matrix for each patient as the product of two latent matrices, but the mapping matrix from those observed medical features to the latent medical concepts is shared over the entire patient population.

When formulating PACIFIER, we enforce sparsity on the latent medical concept mapping matrix to encourage representative and interpretable medical concepts. We also enforce temporal smoothness on the concept value evolution matrix that captures the continuous nature of the patients. We develop an efficient *Block Coordinate Descent* (BCD) scheme for both formulations, that has the capability of processing large-scale datasets. We validate the effectiveness of our method in two real world case studies on predicting the onset risk of Congestive Heart Failure (CHF) patients and End State Renal Disease (ESRD) patients. Our results show that the average prediction AUC in both tasks can be improved significantly (from 0.689 to 0.816 on CHF prediction, and from 0.756 to 0.838 on ESRD respectively) with PACIFIER.

The rest of this paper is organized as follows: Section 2 presents the general representation of EMR and the problem of patient risk prediction which is one important problem that patient phenotyping will be applied to. In Section 3 we introduce the details of PACIFIER. The experimental results are presented in Section 4. In Section 5 we discuss the connection of the proposed approaches to related work and insights for future works. Section 6 concludes the paper.

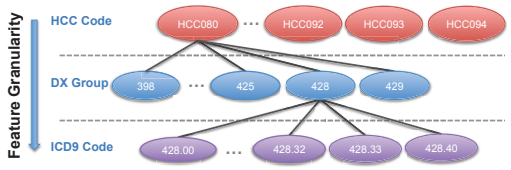


Figure 2: Granularity of medical features. For diagnosis events, features can be constructed at different levels of granularity: ICD9 code, diagnosis code (DxGroup) and HCC code.

2. PATIENT RISK PREDICTION WITH ELECTRONIC MEDICAL RECORDS

Risk prediction is among the most important applications in clinical decision support systems and care management systems, where it often requires building predictive models for a specific disease condition. As Electronic Medical Records (EMR) data becomes widely available, informative features for risk prediction can be constructed from EMR. Based on the EMR data, for example, care providers usually want to assess the risk scores of a patient developing different disease conditions, such as congestive heart failure [30, 6], diabetes [24], and end stage renal disease [1]. Once the risk of a patient is predicted, proper intervention and care plan can be designed accordingly.

The detailed EMR data documents the patient events in time, which typically includes diagnosis, medication, and clinical notes. The diagnosis events are among the most structured, feasible and informative events, and are prime candidates for constructing features for risk prediction [20, 26]. The diagnosis events, often in the form of International Classification of Diseases 9 (ICD9) codes, also come with well-defined feature groups at various levels of granularity such as diagnosis group (DxGroup) and higher-level hierarchical condition categories (HCC). For example, the code 401.1 *Benign Hypertension* belongs to DxGroup 401 *Essential Hypertension*, which is a subcategory in HCC 091 *Hypertension*.

One of the key steps of risk prediction from EMR is to construct features vectors from EMR events, which are used as inputs for classifiers. The goal of feature construction is to capture sufficient clinical nuances that are informative to a specific risk prediction task. Traditionally the feature vectors are directly derived from the raw EMR records [30, 27, 25]. In this paper for each patient we first construct a *longitudinal patient matrix*, with a feature dimension and a time dimension [27]. Maintaining the time dimension enables us to leverage the temporal information of the patients during feature construction. We present the procedure of constructing feature vectors via longitudinal patient matrices as follows.

In a cohort for a disease study, each patient is also associated with a disease status date called *operation criteria date* (OCD), on which the disease is diagnosed. A typical risk prediction task is to predict the disease status of the patients at a certain time point in the future (e.g., half a year). We call this period as the *prediction window*. To build useful predictive models, a prediction window before the OCD is usually specified, and the records before the prediction window are used to train the models, i.e., all records within the prediction window before the OCD are considered to be

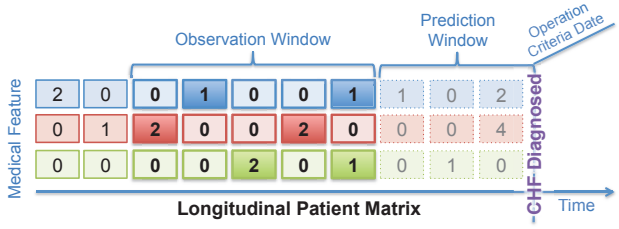


Figure 3: Construction of the longitudinal patient matrix [27] from Electronic Medical Records (EMR). The goal is to predict disease status of a patient at the operation criteria date (OCD), given the past medical information before the prediction window. For each patient, we construct a longitudinal patient matrix, using medical features at a specific granularity. For each patient, the feature vector for classification/regression is finally generated by extracting summary statistics from the longitudinal matrix within the observation window.

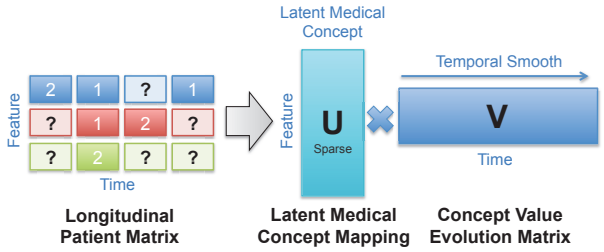


Figure 4: Illustration of the Pacifier framework. We treat a longitudinal patient matrix as a partially observed matrix from a *complete patient matrix*. We assume the medical features can be mapped to some latent medical concepts with a much lower dimensionality such that each medical concept can be viewed as a combination of several observed medical features. For each patient, the values of those medical concepts evolve smoothly over time. Thus the complete patient matrix for each patient can be factorized into a latent medical concept mapping matrix and a concept value evolution matrix.

invisible. Figure 3 illustrates the raw EMR data, OCD, and prediction window.

The next step is to construct a longitudinal patient matrix for each patient from the available EMR events, which consists of two dimensions: the feature dimension and the time dimension. One straightforward way to construct such matrices is to use the finest granularity in both dimensions: use the types of medical events as the feature space for the feature dimension and use *day* as the basic unit for time dimension. Unfortunately the patient matrices constructed in this way are too sparse to be useful. As a remedy, we use *weekly* aggregated time, and the value of each medical feature at one time point is given by the counts of the corresponding medical events within that week. Recall that the medical features can be retrieved at different levels of granularity, which also moderately reduces some sparsity in the data. The choice of feature granularity should not be too coarse, otherwise predictive information within features at a finer level may be lost during the retrieval, as we will show in the experiments. Note that after these preprocessing steps, the constructed patient matrices are still very sparse.

Finally we need to extract summary statistics from the longitudinal patient matrices as the feature vectors for classifiers. Since patients have different lengths of records, typically an *observation window* of interest is defined and the summary statistics (e.g., mean, standard deviation) are extracted within the observation window for all patients. The overall process is given in Figure 3.

3. TEMPORAL DENSIFICATION VIA PACIFIER

During the aforementioned feature construction process, there are many zeros in the longitudinal patient matrices due to the extreme sparsity in the raw EMR data. However, many of these zeros are not real zeros and instead, they indicate missing information (i.e, no visit). Treated as informative values in the feature extraction process, these values are likely to bias the training of classifiers and yield suboptimal performance. In this paper we propose to treat the zeros in the longitudinal patient matrices as missing values, and we densify the sparse matrices before extracting features to reduce the bias introduced by the sparsity, in hopes of that, the densified matrices provide better phenotyping of patients. We propose novel frameworks of densifying the partially observed longitudinal patient matrices, leveraging their observed medical histories. The proposed framework explores the latent structures on both feature and time dimensions and encourages the temporal smoothness of each patient.

Let there be n patients with EMR records available in the cohort, and there be in total p medical features. After the feature construction process we obtain n longitudinal patient matrices with missing entries, one for each patient. For the ith patient, its time dimension is denoted by t_i, i.e., there are medical event records covering a time span of t_i before the prediction window. We denote the ground truth matrix of the ith patient as $X_{(i)} \in \mathbb{R}^{p \times t_i}$, and in our medical records we only have a partial observation of the matrix at some locations, whose indices are given by a set $\Omega_{(i)}$. According to the macro phenotype assumption, we assume the medical features can be mapped to some latent medical concepts space with a much lower *latent dimension* of size k, such that each medical concept can be viewed as a combination of several observed medical features.

Specifically, we assume that the full longitudinal patient matrix can be approximated by a low rank matrix $X_{(i)} \approx U_{(i)}V_{(i)}$, which can be factorized into a sparse matrix $U_{(i)} \in \mathbb{R}^{p \times k}$ whose columns provide mappings from medical features to medical concepts, and a dense matrix $V_{(i)} \in \mathbb{R}^{k \times t_i}$ whose rows indicates the temporal evolution of these medical concepts acting on the patient over time. We call $U_{(i)}$ the *latent medical concept mapping matrix* (abbr. *latent mapping matrix*) and $V_{(i)}$ the *concept value evolution matrix* (abbr. *evolution matrix*). For each patient we assume that the values of those medical concepts evolve smoothly over time. Given the values and locations of observed elements in the longitudinal patient matrices, our proposed densification method learns their latent mapping matrices and evolution matrices. We call this densification framework

PACIFIER, which stands for PAtient reCord densIFIER. The idea of PACIFIER is illustrated in Figure 4.

Based on different natures of the medical cohorts, homogeneous or heterogeneous, we propose two densification formulations: an individual basis approach for heterogeneous patients and a shared basis approach for homogeneous patients, and then we provide an efficient optimization algorithm for PACIFIER that can be used to solve large-scale problems. Here and later we abuse the word *basis* to denote the columns of a concept mapping matrix, while we don't require them to be orthonormal. Note that the *real basis* of the space spanned by the columns of the latent mapping matrix can always be obtained by performing QR factorization on this basis matrix U_i.

3.1 Individual Basis Approach for Heterogeneous Cohort

In the heterogeneous cohort where patients are very different from each other in nature, the medical concepts for each patient may also be different from one patient to another. In the individual basis approach (PACIFIER-IBA), we allow patients to have different latent medical concepts.

Let $\Omega^{\mathsf{c}}_{(i)}$ denote the complement of $\Omega_{(i)}$. We adopt the projection operator $\mathcal{P}_{\Omega_{(i)}}(X_{(i)})$ used in matrix completion [2]: $\mathcal{P}_{\Omega_{(i)}}(X_{(i)}) = X_{(i)}(j,k)$ if $(j,k) \in \Omega_{(i)}$ and $\mathcal{P}_{\Omega_{(i)}}(X_{(i)}) = 0$ otherwise. An intuitive approach for formulating PACIFIER-IBA is to solve the following problem for each patient:

$$\min_{U_{(i)}\geq 0, V_{(i)}} \frac{1}{2t_i}\|\mathcal{P}_{\Omega_{(i)}}(U_{(i)}V_{(i)} - X_{(i)})\|_F^2 + \mathcal{R}(U_{(i)}, V_{(i)}) \quad (1)$$

where $\mathcal{R}(U_{(i)}, V_{(i)})$ denotes the regularization terms that encode our assumptions and prevent overfitting. We also impose a non-negative constraint on the medical concept $U_{(i)}$ because most medical events and measurements in EMR are non-negative, and meaningful medical concepts consist of these medical events should also be non-negative. We now discuss how to design proper terms in $\mathcal{R}(U_{(i)}, V_{(i)})$ that lead to some desired properties:

1) Sparsity. We want only a few significant medical features to be involved in each medical concept so that the concepts can be interpretable. Therefore, we introduce sparsity in the latent mapping matrix $U_{(i)}$ via sparse inducing ℓ_1-norm on $U_{(i)}$. Indeed the non-negativity constraint may have already brought a certain amount of sparsity, and it has been shown that for non-negative matrix factorization, the sparsity regularization can further improve the decomposition [10].

2) Overfitting. To overcome overfitting we introduce an ℓ_2 regularization on the concept value evolution matrix $V_{(i)}$. It can be shown that this term also improves the numerical condition of computing a matrix inversion in our algorithm.

3) Temporal smoothness. A patient matrix describes the continuous evolution of medical features for a patient over time. Thus, along the time dimension it makes intuitive sense to impose the temporal smoothness, such that the value of one column of a longitudinal patient matrix is close to those of its previous and next columns. To this end, we introduce the temporal smoothness regularization on the columns of the concept value evolution, which describes the smooth evolution on the medical concepts. One commonly used strategy to enforce temporal smoothness is via penalizing pairwise difference [37, 34]:

$$\|V_{(i)}R_{(i)}\|_F^2 = \sum\nolimits_{j=1}^{t_i-1}(V_{(i)}(:,j) - V_{(i)}(:,j+1))^2$$

where $R_{(i)} \in \mathbb{R}^{t_i \times t_i+1}$ is the temporal smoothness coupling matrix defined as follows: $R_{(i)}(j,k) = 1$ if $i = j$, $R_{(i)}(j,k) = -1$ if $i = j+1$, and $R_{(i)}(j,k) = 0$ otherwise.

In the loss function of Eq. (1) we want the values of the low-rank matrix to be close to $X_{(i)}$ at the observed locations, directly solving which may lead to complex algorithms. An alternative way is to introduce an intermediate matrix S_i such that $\mathcal{P}_{\Omega_i}(S_i) = \mathcal{P}_{\Omega_i}(X_i)$, and we want $U_{(i)}V_{(i)}$ to be close to $S_{(i)}$. An immediate advantage of propagating the observed information from $X_{(i)}$ to $U_{(i)}V_{(i)}$ indirectly is that we can derive very efficient algorithms and data structures, which give the capability of solving large-scale problems, as we will show later. To this end, we propose the following PACIFIER-IBA learning model for each patient:

$$\begin{aligned}\min_{S_{(i)},U_{(i)},V_{(i)}} &\tfrac{1}{2t_i}\|S_{(i)} - U_{(i)}V_{(i)}\|_F^2 + \lambda_1\|U_{(i)}\|_1 \quad (2)\\ &+\lambda_2\tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3\tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2,\\ \text{subject to:}\quad &\mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}), U_{(i)} \geq 0\end{aligned}$$

3.2 Shared Basis Approach for Homogeneous Cohort

In homogeneous cohorts where the medical concepts of patients are very similar to each other, we can assume that all patients share the same medical concept mapping $U \in \mathbb{R}^{p\times k}$. We propose the following PACIFIER-SBA formulation:

$$\begin{aligned}\min_{\{S_{(i)}\},U,\{V_{(i)}\}} &\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|S_{(i)} - UV_{(i)}\|_F^2 + \lambda_1\|U\|_1 \quad (3)\\ &+\lambda_2\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2\\ \text{subject to:}\quad &\mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}), U \geq 0\end{aligned}$$

Since the densification of all patients are now coupled via the shared concept mapping, an immediate benefit of the PACIFIER-SBA formulation is that, we can transfer some knowledge among the patients, which is attractive especially when the available information for each patient is very limited and the patients are homogeneous in nature. We demonstrate in the experiments that the PACIFIER-SBA performs better than IBA when patients are homogeneous.

On a separate note, considering densification of each patient as a learning task, the SBA approach performs inductive transfer learning among the tasks via a shared representation of U and thus belongs to the multi-task learning paradigm [7, 8]. As such, the SBA in nature is a multi-task matrix completion problem.

3.3 Optimization Algorithm

The formulations of PACIFIER are non-convex and we present a block coordinate descent (BCD) optimization algorithm to obtain a local solution. Note that for each patient the subproblem of PACIFIER-IBA in Eq. (2) is a special case of the problem of PACIFIER-SBA in Eq. (3) given $n = 1$. Therefore in this section we present the algorithm for Eq. (3).

1) Solve U^+ given $V^-_{(i)}$ and $S^-_{(i)}$:

$$U^+ = \operatorname*{argmin}_{U\geq 0}\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|S^-_{(i)} - UV^-_{(i)}\|_F^2 + \lambda_1\|U\|_1. \quad (4)$$

This is a standard non-negative ℓ_1-norm regularized problem and can be solved efficiently using scalable first order methods such as spectral projected gradient [29] and proximal Quasi-Newton method [14].

2) Solve $V_{(i)}^+$ given U^+ and $S_{(i)}^-$:

$$\{V_{(i)}^+\} = \underset{\{V_{(i)}\}}{\operatorname{argmin}} \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|S_{(i)}^- - U^+V_{(i)}\|_F^2 + \lambda_2 \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3 \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2 \quad (5)$$

Note that the terms are decoupled for each patient, resulting in a set of minimization problems:

$$V_{(i)}^+ = \underset{V_{(i)}}{\operatorname{argmin}} \tfrac{1}{2}\|S_{(i)}^- - U^-V_{(i)}\|_F^2 + \tfrac{\lambda_2}{2}\|V_{(i)}\|_F^2 + \tfrac{\lambda_3}{2}\|V_{(i)}R_{(i)}\|_F^2, \quad (6)$$

The problem in (6) can be solved using existing optimization solvers. Moreover, since the problem is smooth, it admits a simple analytical solution [37].

LEMMA 1. *Let $Q_1\Lambda_1Q_1^T = U^TU + \lambda_2 I$ and $Q_2\Lambda_2Q_2^T = \lambda_3 R_{(i)}R_{(i)}^T$ be eigen-decompositions, and let $D = Q_1^TU^TS_{(i)}Q_2$, the problem (6) admits an analytical solution:*

$$V_{(i)}^* = Q_1\hat{V}Q_2, \quad \textit{where } \hat{V}_{j,k} = \tfrac{D_{j,k}}{\Lambda_1(j,j)+\Lambda_2(k,k)}. \quad (7)$$

Note that the parameter λ_2 improves the stability of the 'inversion' in $V_{j,k}$ so that the denominator is guaranteed to be a positive number. Excluding the time of the two QR factorizations, the cost of computing the analytical form solution for each sample is given by $O(k^2pt)$. The computation can be greatly accelerated as shown in the next section. Including the time of QR factorizations, obtaining the results from the analytical form is typically 100 times faster than that of solving (5) using optimization solvers.

3) Solve $S_{(i)}^+$ given U^+ and $V_{(i)}^+$:

$$\{S_{(i)}^+\} = \underset{\{S_{(i)}\}}{\operatorname{argmin}} \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|S_{(i)} - U^+V_{(i)}^+\|_F^2. \quad (8)$$
$$\text{subject to:} \quad \mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$$

The problem is a constrained Euclidean projection, and is decoupled for each $S_{(i)}^+$. The subproblem for each one admits a closed-form solution: $S_{(i)}^+ = \mathcal{P}_{\Omega_{(i)}^c}(U^+V_{(i)}^+) + \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$.

Algorithm 1 The BCD algorithm for solving the PACIFIER-SBA in formulation (3). Given $n = 1$, the algorithm also solves the PACIFIER-IBA for each patient in the formulation (2).

Input: Observed locations $\{\Omega_{(i)}\}_1^n$, values of the observed entries for each patient $\{\mathcal{P}_{\Omega_{(i)}}(X_{(i)})\}_1^n$, initial solutions $\{V_{(i)}^0\}_1^n$, sparse parameter λ_1, parameter λ_2, smooth parameter λ_3, latent factor k.
Output: U^+, $\{V_{(i)}^+\}_1^n$, $\{S_{(i)}^+\}_1^n$.

```
Set V_(i)^- = V_(i)^0, S_(i)^- = P_Ω(i)(X_(i)) for all i.
while true do
    Update U^+ by solving (4) via ℓ1 solvers (e.g. [14, 29]).
    Update V_(i)^+ by computing (7).
    Update S_(i)^+ = P_Ω(i)^c(U^+V_(i)^+) + P_Ω(X_(i))
    if U^+ and {V_(i)^+}_1^n converge then
        return U^+ and {V_(i)^+}_1^n
    end if
    Set V_(i)^- = V_(i)^+ and S_(i)^- = S_(i)^+ for all i.
end while
```

We summarize the BCD algorithm of PACIFIER-SBA in Algorithm 1. In our implementation, we randomly generate the initial concept evolution matrix $V_{(i)}^0$, and set $U_{(i)}^0 = (0)$. Therefore the initial value of $S_{(i)}^-$ is given by $S_{(i)}^- = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}) + \mathcal{P}_{\Omega_{(i)}^c}(\mathbf{0}V_{(i)}^0) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$. Since the problem of PACIFIER is non-convex, and thus it is easy to fall into a local minimum. One way to escape from local minimum is to 'restart' the algorithm by slightly perturbing V_i after the algorithm converges, and compute a new solution. Among the many solutions, we use the one with the lowest function value. In the following section we discuss how to accelerate the algorithm to solve large-scale problems.

3.4 Efficient Computation for Large Scale Problems

For large scale problems, the storage of the matrix S_i and $O(d^2)$-level computations are prohibitive. However, we notice that in each iteration, we have that $S_{(i)}^+ = \mathcal{P}_{\Omega_i^c}(U^+V_{(i)}^+) + \mathcal{P}_{\Omega_i}(X_{(i)}) = U^+V_{(i)}^+ + \mathcal{P}_{\Omega_i}(X_{(i)} - U^+V_{(i)}^+)$. The "low rank + sparse" structure of $S_{(i)}^+$ indicates that there is no need to store the full matrices. Instead we only need to store two smaller matrices depending on k and a sparse residual matrix $\mathcal{P}_{\Omega_i}(X_{(i)} - U^+V_{(i)}^+)$. This structure can be used to greatly accelerate the computation of Eqs. (4) and (5). In the following discussion we denote $S_{(i)} = U_{S_{(i)}}V_{S_{(i)}} + S_{S_{(i)}}$.

1) Solve U. The major computational cost of Eq. (4) lies on the evaluation of the loss function and the gradient of the smooth part. Taking advantage of the structure of S_i. We show that all prohibitive $O(d^2)$ level operations can be avoided given the special structures of $S_{(i)}^+$.

Gradient Evaluation:

$$\begin{aligned}
&\nabla_U \left(\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|S_{(i)} - UV_{(i)}\|_F^2\right)\\
=&\nabla_U \left(\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|(U_{S_{(i)}}V_{S_{(i)}} + S_{S_{(i)}}) - UV_{(i)}\|_F^2\right)\\
=&\sum\nolimits_{i=1}^{n} \tfrac{1}{t_i}\left(U(V_{(i)}V_{(i)}^T) - U_{S_{(i)}}(V_{S_{(i)}}V_{(i)}^T) - S_{S_{(i)}}V_{(i)}^T\right)
\end{aligned}$$

Objective Evaluation:

$$\begin{aligned}
&\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\|S_{(i)} - UV_{(i)}\|_F^2\\
=&\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\operatorname{tr}(S_{(i)}^TS_{(i)} - 2S_{(i)}^TUV_{(i)} + V_{(i)}^TU^TUV_{(i)})\\
=&\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i}\Big(\operatorname{tr}\left(V_{S_{(i)}}^T(U_{S_{(i)}}^TU_{S_{(i)}}V_{S_{(i)}})\right) + \operatorname{tr}\left(S_{S_{(i)}}^TS_{S_{(i)}}\right)\\
&+ 2\operatorname{tr}\left((S_{S_{(i)}}^TU_{S_{(i)}})V_{S_{(i)}}\right) + \operatorname{tr}\left(V_{(i)}^T(U^TUV_{(i)})\right)\\
&- 2\operatorname{tr}\left(V_{S_{(i)}}^T(U_{S_{(i)}}^TUV_{(i)})\right) - 2\operatorname{tr}\left((S_{S_{(i)}}^TU)V_{(i)}\right)\Big)
\end{aligned}$$

For the evaluation of the loss function, it can be shown that the complexity is $O(k^2npt)$ if all patients have t time slices. Similarly the complexity of computing the gradient is also given by $O(k^2npt)$. Therefore in the optimization, the computational cost in each iteration is linear with respect to n, p and t. Thus the algorithm is scalable to large data.

2) Solve V. The term $U^TS_{(i)}$ can again be computed efficiently using the similar strategy as above. Recall that in solving $V_{(i)}^+$ we need to perform eigen-decomposition on two matrices: a $\mathbb{R}^{k\times k}$ matrix U^TU and a $\mathbb{R}^{t\times t}$ tridiagonal matrix $R_{(i)}R_{(i)}^T$. The two matrices are equipped with special structures: the matrix U^TU is a low-rank matrix, and the matrix $R_{(i)}R_{(i)}^T$ is a tridiagonal matrix (a very sparse matrix), whose eigen-decomposition can be solved efficiently.

Note that the complexity of time dimension is less critical, because that in most EMR cohorts, the time dimension of the patients are often less than 1000. Recall that the finest time unit of the EMR records is day. Using weekly granularity, 1000 time dimension covers up to 20 years of records. In our implementation we use the built-in eigen-decomposition of Matlab, which typically takes less than 1 sec for a matrix with a time dimension of 1000 on regular desktop computers.

3.5 Latent Dimension Estimation

In the formulations in Eq. (2) and Eq. (3), we need to estimate the latent dimension of the patient matrices. Indeed, we can choose the latent dimension via validation methods, as done for other regularization parameters. As an alternative, we can use the rank estimation heuristic to adaptively set the latent dimension of the matrices by inspecting the information in the QR decomposition of the latent concept mapping matrix U, assuming that the latent dimension information of all patients is collectively accumulated in U after a few iterations of updates. The idea was originally proposed in [28, 23] to estimate the rank during the matrix completion of a single matrix.

In order to be self-contained we briefly summarize the algorithm as follows. After a specified iterations of updates, we perform the economic QR factorization on $UE = Q_U R_U$, where E is a permutation matrix such that $|\operatorname{diag}(R_U)| := [r_1 \dots r_k]$ is non-increasing after the permutation. Denote $\mathcal{Q}_p = r_p / r_{p+1}$, and $\mathcal{Q}_{\max} = \max(\mathcal{Q}_p)$, and the location is given by $p_{\max}$. We compute the following ratio:

$$\tau = \frac{(K-1)\mathcal{Q}_{\max}}{\sum_{\{p \neq p_{\max}\}} \mathcal{Q}_i}.$$

A large τ indicates a large drop in the magnitude of $\mathcal{Q}_i$ after $p_{\max}$ elements, and we thus reduce the latent factor k to $p_{\max}$, retaining only the first $p_{\max}$ columns of U and the corresponding rows of the evolution matrices $\{V_{(i)}\}$. In our implementation we only perform the estimation once. Empirically as shown in Section 4.2, the latent dimension estimation works well when the Pacifier-Sba works, i.e., patients are homogeneous, sharing a few latent concepts.

In the Iba approach the completion of patients are independent. If we apply latent dimension estimation on each patient, then each patient matrix may have a latent dimension different from others. This imposes difficulties when it comes to analyze the patients, and thus the estimation is not used in Iba.

4. EMPIRICAL STUDY

In this section we present the experimental results to demonstrate the performance of the proposed Pacifier methods Iba and Sba. We then study the scalability of the proposed algorithm with varying feature dimensions, time dimensions, sample sizes, latent dimensions, and ratios of the observed entries. We then apply the proposed Pacifier framework on two real clinical cohorts to demonstrate the improvement on predictive performance achieved by our approaches. The code for the proposed algorithm is available in [33].

4.1 Scalability

In this section we study the scalability of the proposed algorithm using synthetic datasets. In each of the following studies, we generate random datasets with a specified sample n, feature dimension p, average time dimension t, latent dimension k, and observation density $\|\Omega_i\|$. For simplicity we let all samples have the same time dimension. We report the average time cost over 50 iterations. For the two algorithms we set all parameters to be $1e-8$ in all studies.

Sample Size. We fix $p = 100$, $t = 100$, $r = 10$, $\|\Omega_i\| = 0.01$, and vary the sample size $n = 200 : 200 : 1800$. The results are given in Figure 5(a). We observe that for both methods the time costs increase linearly with respect to the sample size. The cost of Iba grows faster than the Sba version, which is expected because in Iba the computation costs of the loss and the gradients are more than those of Sba.

Feature Dimension. We fix $n = 100$, $t = 100$, $r = 10$, use $\|\Omega_i\| = 0.01$, and vary the feature dimension $p = 200 : 200 : 1800$. The results are given in Figure 5(b). We see that the time costs for both methods increase linearly with respect to feature dimension, which is consistent with our complexity analysis. The linear complexity of feature dimension is desired in clinical applications, since one might want to use as much information available as possible, resulting in a large feature space.

Time Dimension. We fix $n = 100$, $p = 100$, $r = 10$, $\|\Omega_i\| = 0.01$, and vary the time dimension $t = 100 : 100 : 900$. The results are given in Figure 5(c). We find superlinear complexity on the time dimension for both methods, which mainly comes from the eigen decomposition. The complexity on time dimension is less critical in the sense that for most medical records and longitudinal study, the time dimension is very limited. For example, if the time granularity is weekly, then we have 52 time dimensions each year. If 20-year records are available for one patient, then it yields only 1040 time dimensions. Besides, the eigen decomposition can be implemented in the way that utilizes the extreme sparsity of the temporal smoothness coupling matrix.

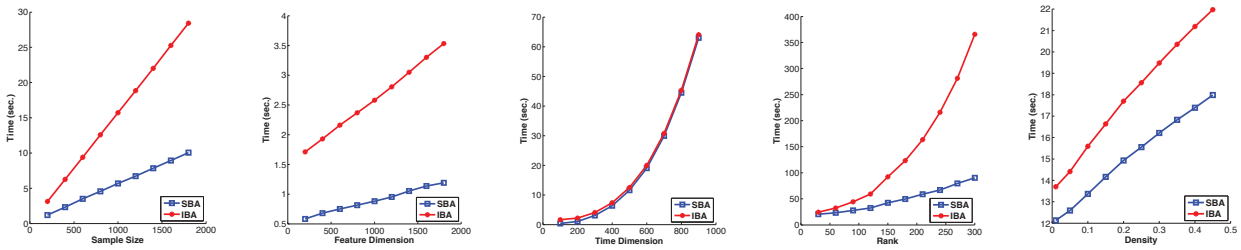
Latent Dimension. We fix $n = 100$, $p = 500$, $t = 500$, $\|\Omega_i\| = 0.01$, and vary the latent dimension input of the algorithms $r = 20 : 20 : 160$. The results are given in Figure 5(d). We find that the time costs increase superlinearly with respect to latent dimension for both methods, and the complexity of Sba is close to be linear.

Observed Entries. We fix $n = 100$, $p = 1000$, $t = 500$, $r = 10$, and vary the percentage of the observed entries $\|\Omega_i\| = 0.05 : 0.05 : 0.45$. The results are given in Figure 5(d). We see that the time costs increase only sublinearly with respect to the set of observed entries.

We note that the complexity of Pacifier-Iba is of the same order as that of Sba. The difference between the two methods comes from the computation of the objective value and gradient in the U step. It is obvious that the IBA methods can be parallelized because the computation of all samples are decoupled. Similarly, the major computational complexity of SBA comes from the computation of U in the optimization and eigen-decomposition of $V_{(i)}$, which can also be parallelized by segmenting the computation of each patient.

4.2 Predictive Performance on Real Clinical Cohorts

To gauge the performance of the proposed Pacifier framework we apply the two formulations on two real EMR cohorts from one of our clinical partners. In one cohort we



(a) Sample Size (b) Feature Dimension (c) Time Dimension (d) Latent Dimension (e) Observed Entries

Figure 5: Studies of scalability of Pacifier-Iba and Pacifier-Sba. In each study we vary one of the scale factors while fix other factors, and record the time costs. Both methods have the same complexity: linear with respect to samples size and feature dimension; superlinear with respect to time dimension and latent dimension; sublinear with respect to the number of observed entries.

study the predictive modeling of congestive heart failure (CHF), and in the other cohort we study end stage renal disease (ESRD). In both EMR cohorts we are given a set of patients associated with their outpatient diagnosis events in ICD9 codes and the corresponding timestamps. In our experiments we use the prediction windows lengths suggested by physicians (180 days for CHF and 90 days for ESRD), and we remove all events within the prediction window before the operation criteria date.

To construct the longitudinal patient matrices to be imputed, we use EMR data at the weekly granularity as discussed in Section 2. We select the patients with more than 100 events. Note that we are working on a large feature dimension, and thus for a patient with 100 EMR events the longitudinal patient matrix is still extremely sparse. Note that in our cohorts the number of case patients is much smaller than control patients, which is very common in most clinical studies. To avoid the effects of biased samples, we perform random under-sampling on the control patients so that we have the equal number of case and control patients in our datasets. To this end, we have constructed two datasets: 1) CHF dataset with 249 patients in each class; 2) ESRD dataset with 187 patients in each class.

The raw feature space in the low-level ICD9 codes is 14313. Because the matrix constructed using the low-level ICD9 codes is too sparse, we retrieve the medical features at coarser granularities. In order to study the effects of features at different granularities, we compare the medical features at ICD9 diagnosis group level (DxGroup) and HCC level. At DxGroup level there are 1368 features and at HCC level there are 252 features. In the two studies we consider the following commonly-used baselines methods:

- Zero Imputation (RAW). An intuitive way to impute missing values, which is equivalent to mean value imputation when the data set is first normalized (zero mean and unit standard deviation). This method is standard in the current medical literature for clinical studies [25, 27, 30].
- Row Average (AVG). In this baseline approach we fill the missing value using the average value of the observed values of the feature over time.
- Interpolation (INT) [5]. We use the next observation and previous observation along the timeline to interpolate the missing elements.
- Next Observation Carry Backward (NOCB) [5]. Missing values are filled using the next observation of this medical feature along the timeline.
- Last Observation Carry Forward (LOCF) [5]. Missing values are filled using the previous observation of this medical feature along the timeline.

We compare the baseline methods with the following competing methods:

- Individual Basis Pacifier (Iba). Each patient is densified using Algorithm 1.
- Iba without temporal smoothness (Iba-NT). This variant of Pacifier-IBA sets the temporal regularization λ_3 to 0.
- Shared Basis Pacifier (Sba) using Algorithm 1.
- Sba without temporal smoothness (SbaNT). This variant of Pacifier-Sba sets the temporal regularization λ_3 to 0.
- Sba with Latent Dimension Estimation (Sba-E). The latent dimension estimation is described in Section 3.5, and only used once during the algorithm.
- Sba without Temporal Smoothness and with Latent Dimension Estimation (SbaNT-E). This variant of Pacifier-Sba sets the temporal regularization λ_3 to 0 and uses latent dimension estimation once.

Note that for the extremely sparse matrix as the clinical data in our studies, classical imputation methods such as those based on k-nearest neighbor [9] and expectation maximization [22] do not work. The methods IbaNT and SbaNT are included in the study to explore the effectiveness of the proposed temporal smoothness. For the parameter estimation we have separated an independent set of samples for validation, and we select the parameters that give the lowest recovery error on the validation set. In Iba, Sba, and SbaNT, the latent dimension k is also determined via the validation set.

We finally test the predictive performance on the completed datasets using sparse logistic regression classifier (we use the SLEP implementation [15]). From the completed datasets, we derive features by averaging the features along the time dimension within a given observation window (52 weeks). To this end, each patient is represented as a vector of the same dimension as the feature dimension. We then randomly split the samples into 90% training and 10% testing, and train the classifier on the training data. The classifier parameter is tuned using standard 10 fold cross validation. We repeat the random splitting for 20 iterations, and report the average performance over all iterations. In order to be comparable, the splitting is the same for all methods in each iteration.

CHF Cohort. The predictive performance of competing methods is presented in Table 1. We find that in the CHF cohort: 1) most of the proposed Pacifier approaches and their variants significantly improve the predictive performance as

Table 1: Predictive performance on the CHF cohort using DxGroup and HCC features.

DxGroup Features			
Method	AUC	Sensitivity	Specificity
RAW	0.689 ± 0.058	0.747 ± 0.046	0.528 ± 0.115
AVG	0.671 ± 0.051	0.744 ± 0.064	0.482 ± 0.083
INT	0.644 ± 0.066	0.803 ± 0.062	0.468 ± 0.110
NOCB	0.658 ± 0.048	0.845 ± 0.073	0.443 ± 0.096
LOCF	0.689 ± 0.055	0.866 ± 0.082	0.456 ± 0.087
IBA	$\mathbf{0.816 \pm 0.040}$	$\mathbf{0.843 \pm 0.054}$	$\mathbf{0.657 \pm 0.078}$
IBANT	0.754 ± 0.056	0.762 ± 0.089	0.597 ± 0.097
SBA	0.750 ± 0.062	0.776 ± 0.067	0.640 ± 0.106
SBANT	0.706 ± 0.054	0.672 ± 0.079	0.631 ± 0.066
SBA-E	0.730 ± 0.064	0.695 ± 0.074	0.653 ± 0.095
SBANT-E	0.661 ± 0.073	0.678 ± 0.090	0.588 ± 0.095
HCC Features			
Method	AUC	Sensitivity	Specificity
RAW	0.645 ± 0.089	0.672 ± 0.086	0.529 ± 0.072
AVG	0.660 ± 0.053	0.683 ± 0.063	0.526 ± 0.089
INT	0.596 ± 0.072	0.768 ± 0.093	0.489 ± 0.082
NOCB	0.602 ± 0.081	0.694 ± 0.088	0.511 ± 0.093
LOCF	0.625 ± 0.067	0.852 ± 0.079	0.480 ± 0.083
IBA	$\mathbf{0.755 \pm 0.071}$	0.747 ± 0.085	$\mathbf{0.641 \pm 0.084}$
IBANT	0.727 ± 0.060	0.740 ± 0.087	0.614 ± 0.070
SBA	0.736 ± 0.066	$\mathbf{0.753 \pm 0.089}$	0.629 ± 0.074
SBANT	0.645 ± 0.070	0.686 ± 0.087	0.550 ± 0.095
SBA-E	0.702 ± 0.079	0.688 ± 0.106	0.616 ± 0.067
SBANT-E	0.669 ± 0.062	0.702 ± 0.082	0.538 ± 0.079

compared to the baseline RAW approach. The best AUC obtained by PACIFIER-IBA dataset is 0.816 while the baseline is only 0.689 (a gain of 0.127); 2) the individual basis approaches outperform shared based ones; 3) temporal regularization significantly improves the predictive performance for all methods; 4) the methods with latent dimension estimation perform worse than those that do not use latent dimension estimation on this cohorts; 5) the features at DxGroup level outperform HCC level, which might be due to that in this predictive task, a fine granularity is likely to maintain more predictive information, than a coarse one.

ESRD Cohort. The predictive performance on ESRD cohort is given in Table 2. For the DxGroup features we observe similar patterns that is, IBA outperforms all other methods, which achieves an AUC of 0.828, compared to the baseline RAW method that achieves 0.756 (a gain of 0.072). The variants with temporal smoothness perform much better than the ones without temporal smoothness. For the HCC features we see that: 1) the shared basis approaches perform as well as the independent basis, where SBA-E achieves an AUC of 0.827. 2) again the temporal smoothness significantly improves the performance. 3) latent dimension estimation works well and outperforms the ones without latent dimension estimation.

As a summary, the experimental results have demonstrated the effectiveness of the proposed methods on real clinical data, and the temporal smoothness regularization brings significant improvements on predictive performance. In real clinical data, the samples tend to be heterogeneous and therefore the independent basis approaches perform better. However, using the HCC features of the two datasets, shared basis approaches perform better than using the DxGroup features. One potential explanation is that, using HCC features where the features space is smaller and features themselves are coarser (in terms of clinical concepts), the patients tend to be more homogeneous. We also notice that the latent dimension estimation only works well when shared basis works well. Recall that the idea of latent dimension estimation is to detect the jumps in the diagonal elements from the R_U factor of QR factorization. This is expected because

Table 2: Predictive performance on the ESRD cohort with DxGroup and HCC features.

DxGroup Features			
Method	AUC	Sensitivity	Specificity
RAW	0.756 ± 0.086	0.831 ± 0.113	0.581 ± 0.077
AVG	0.775 ± 0.079	0.821 ± 0.093	0.592 ± 0.084
INT	0.747 ± 0.083	0.919 ± 0.104	0.568 ± 0.110
NOCB	0.766 ± 0.092	0.914 ± 0.099	0.556 ± 0.103
LOCF	0.787 ± 0.085	0.958 ± 0.107	0.577 ± 0.079
IBA	$\mathbf{0.838 \pm 0.072}$	$\mathbf{0.842 \pm 0.099}$	0.658 ± 0.106
IBANT	0.796 ± 0.066	0.806 ± 0.101	0.600 ± 0.095
SBA	0.811 ± 0.065	0.769 ± 0.091	$\mathbf{0.722 \pm 0.097}$
SBANT	0.763 ± 0.068	0.719 ± 0.109	0.697 ± 0.075
SBA-E	0.803 ± 0.056	0.753 ± 0.098	0.681 ± 0.090
SBANT-E	0.770 ± 0.082	0.689 ± 0.099	0.700 ± 0.110
HCC Features			
Method	AUC	Sensitivity	Specificity
RAW	0.758 ± 0.058	0.747 ± 0.085	0.656 ± 0.093
AVG	0.778 ± 0.055	0.789 ± 0.088	0.660 ± 0.088
INT	0.729 ± 0.067	0.752 ± 0.091	0.652 ± 0.094
NOCB	0.752 ± 0.079	0.775 ± 0.089	0.658 ± 0.095
LOCF	0.771 ± 0.068	0.808 ± 0.082	0.665 ± 0.081
IBA	0.826 ± 0.051	0.800 ± 0.085	0.708 ± 0.080
IBANT	0.802 ± 0.064	0.775 ± 0.094	0.714 ± 0.089
SBA	0.820 ± 0.064	0.789 ± 0.091	$\mathbf{0.722 \pm 0.092}$
SBANT	0.771 ± 0.082	0.733 ± 0.084	0.681 ± 0.102
SBA-E	$\mathbf{0.827 \pm 0.067}$	$\mathbf{0.814 \pm 0.077}$	0.706 ± 0.096
SBANT-E	0.785 ± 0.060	0.736 ± 0.065	0.717 ± 0.092

if the patients are homogeneous and share only a few basis, then obviously there are such natural jumps.

4.3 Macro Phenotypes Learned from Data

In this section we show some meaningful medical concepts learned by the proposed PACIFIER-SBA method. In the latent medical concept mapping matrix U, we are able to obtain feature groups from data, because of the sparsity on the matrix. We first normalize weights of the columns such that the sum of each column is equal to 1. The normalized weights indicate the percentages of medical features contributing to the medical concept. We rank the medical features according to their contributions and find that in most of the medical concepts the top medical features are typically related and are comorbidities of a certain disease. In Figure 3, we show a list of medical concepts obtained from our CHF cohort. For example, in the first medical concept, the highly ranked diagnosis groups are all related to *Cardiovascular Disease*, e.g., Heart failure (428), Hypertension (401) and Dysrhythmias (427), and the second medical concepts include features that are typical related to *Diabetes* and its related comorbidities such as Hypertension (401), Chronic renal failure (585). In the CHF cohort, we have also found very similar medical concepts.

5. RELATED WORKS AND DISCUSSION

In this paper we treat the zeros in the longitudinal patient matrices as missing values, and proposed a novel framework PACIFIER to perform temporal matrix completion via low-rank factorization. To the best of our knowledge, there are no prior work that applies matrix completion techniques to solve the data sparsity in EMR data. The proposed PACIFIER framework aims at densifying the extremely sparse EMR data by performing factorization based matrix completion. The differences between the proposed completion method and existing works are that: instead of treating each patient as vectors and forming a single matrix, we treat each patient as a matrix with missing entries and consider a set of related matrix completion problems. We further propose to

Table 3: Medical concepts discovered by the Pacifier-Sba in our CHF cohort. In each medical concept, we firstly normalize the weights of the medical features in the medical concepts learned and rank the features. For each medical concept we list top 10 medical features and their diagnosis group codes.

Medical Concept: Cardiovascular Diseases		
Weight	DxGrp	Description
0.164	428	Heart failure
0.121	401	Essential hypertension
0.113	427	Cardiac dysrhythmias
0.108	780	General sympt.
0.141	414	Other form of chronic ischemic heart disease
0.053	785	Symp. inv. cardiovascular sys.
0.052	786	Symp. inv. respir. sys. and other chest sympt.
0.046	402	Hypertensive heart disease
0.042	272	Diso. of lipoid metabolism
Medical Concept: Diabetes		
Weight	DxGrp	Description
0.211	250	Diabetes mellitus
0.129	272	Diso. of lipoid metabolism
0.115	278	Obesity and other hyperalign.
0.095	593	Other diso. of kidney and ureter
0.093	585	Chronic renal failure
0.068	599	Other diso. of urethra and urinary tract
0.065	790	Nonspe. find on exam of blood
0.058	401	Essential hypertension
0.023	366	Cataract
0.019	285	Other and unspecified anemias
Medical Concept: Lung Diseases		
Weight	DxGrp	Description
0.117	518	Other diseases of lung
0.112	496	Chronic airways obstruction
0.110	786	Sympt. inv. respir. sys. and other chest symp.
0.098	V72	Special investigations and exam
0.089	493	Asthma
0.087	599	Other diso. of urethra and urinary tract
0.086	466	Acute bronchitis and bronch.
0.078	780	General symp.
0.067	787	Symp. inv. digestive sys.
0.057	793	Nonspec. ab. find on radio. and other exam of body structure
Medical Concept: Osteoarthrosis		
Weight	DxGrp	Description
0.185	729	Other diso. of soft tissues
0.123	715	Osteoarthrosis and allied diso.
0.120	726	Peripheral enthesopathies and allied syndr.
0.118	401	Essential hypertension
0.082	733	Other diso. of bone and cartilage
0.081	366	Cataract
0.069	719	Other and unspec. diso. of joint
0.066	272	Diso. of lipoid metabolism
0.065	780	General symp.
0.008	244	Acquired hypothyroidism
Medical Concept: Disorder of joints and softtissues		
Weight	DxGrp	Description
0.103	719	Other and unspec. diso. of joint
0.096	729	Other diso. of soft tissues
0.081	789	Other symp. involving abdomen and pelvis
0.078	722	Intervertebral disc diso.
0.058	724	Other and unspec. diso. of back
0.056	780	General symp.
0.055	721	Spondylosis and allied diso.
0.053	728	Diso. of muscle, ligament, and fascia
0.048	733	Other diso. of bone and cartilage
0.048	723	Other diso. of cervical region

incorporate the temporal smoothness to utilize the hidden temporal information of each patient.

The problem of imputation via matrix completion problem is one of the hottest topics in data mining and machine learning. In many areas such as information retrieval and social network, the data matrix is so sparse that classical imputation methods does not work well. The basic problem setting of the matrix completion is to recover the unknown data from only a few observed entries, imposing certain types of assumptions on the matrix to be recovered. The most popular assumption is to assume that the matrix has a low rank structure [2, 18, 28, 31]. There are two types of matrix completion in terms of the assumption on the observed entries: The first type assumes that the observation has no noise, and the goal is to find a low rank matrix whose values at the observed locations are exactly the same as the given ones [2, 3, 11]. In real world applications, however, noise is ubiquitous and thus the rigid constraint on the observed locations may result in overfitting. In contrast, the noisy matrix completion methods only require the values at the observed locations to be close to the given data [18, 28]. Directly dealing with the rank function in objectives are shown to be NP-Hard. Therefore many approaches seek to use the trace norm which is the convex envelope of the rank function [2, 11, 18]. Most of these approaches, however, require singular value decomposition (SVD) on large matrices, the complexity of which is prohibitive for large scale problems. Recent years have witnessed surging interests on the local search methods, which seek a local solution with extremely efficient algorithms [21, 28]. The Pacifier framework is among these efficient local approaches, which does not require SVD and can be applied to solving large scale problems.

The completed data for each patient has the factorization form of $X_{(i)} = U_{(i)}V_{(i)}$, and for Sba all patients have the same $U_{(i)}$. Clearly, one advantage of Sba is that we have simultaneously learned a shared low-dimensional feature space for all patients, and their coordinates that can be used as new (and reduced) features. To see this, let $U = Q_U R_U$ be the QR factorization of U, then for each patient we have that $X_{(i)} = UV_{(i)} = Q_U(R_U V_{(i)})$, indicating that rows of $(RV_{(i)})$ can be considered as coordinates on the low dimensional space whose bases are given by columns of Q_U. One issue brought by the shared mapping is that the latent dimension is limited by the lowest time dimension of the patient, i.e., $\min_i t_i > k$. One solution is that we can extend the time dimension of the patients with non-informative time dimensions of all zeros.

We have shown in the experiments that a shared concept mapping works better on homogeneous samples while individual mappings work better on heterogeneous samples. In reality the samples may form some groups such that within the groups the patients are homogeneous and patients from different groups may be heterogeneous. The degree of homogeneous/heterogeneous is also affected by feature granularity as shown in our real clinical experiments, where in finer feature level the patients appear to be more heterogeneous. It is thus interesting to explore how to simultaneously identify feature groups and patient groups to further improve the quality of matrix completion. To do so, we can incorporate group learning into the objective as done in [32]:

$$\min_{\mathcal{G},\{S_i,U_j,V_i\}} \frac{1}{g}\Sigma_{j=1}^{g}\frac{1}{|\mathcal{G}_j|}\Sigma_{i\in\mathcal{G}_j}\|S_i - U_jV_i\|_F^2 + \mathcal{R}(\{U\},\{V\})$$

where $\mathcal{G}$ is the patient group assignment matrix, and patients within each group $\mathcal{G}_j$ share the same basis U_j. We leave this interesting study to our future works. One final note – the proposed Pacifier framework proposed in this paper is not limited to healthcare domain, they can also be applied to temporal collaborative filtering [12, 16, 31], where each user has a rating preference that changes overtime.

6. CONCLUSION

In this paper, we propose a data driven phenotyping framework called Pacifier (PAtient reCord densIFIER) to den-

sify the sparse EMR data. The PACIFIER interprets the longitudinal EMR of each patient as a sparse matrix with a feature dimension and a time dimension, and estimates the missing entries in those matrices by leveraging the latent structures on both time and feature dimensions. We propose two formulations: Individual Basis Approach (IBA), which densifies the matrices patient by patient, and Shared Basis Approach (SBA), which densifies the matrices of a group of patients jointly. We develop an efficient optimization algorithm to solve the framework, which scales to large-size datasets. We have performed extensive empirical evaluations on both synthetic and real datasets, including two real world clinical datasets. Our results show that the predictive performance in both tasks can be improved significantly after the densification by the proposed methods.

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by NIH R01 LM010730, NSF IIS-0953662, MCB-1026710, and CCF-1025177.

7. REFERENCES

[1] J. Blacher, A. P. Guerin, B. Pannier, S. J. Marchais, and G. M. London. Arterial calcifications, arterial stiffness, and cardiovascular risk in end-stage renal disease. *Hypertension*, 38(4):938–942, 2001.

[2] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. on Opt.*, 20(4):1956–1982, 2010.

[3] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Found. of Comp. Math.*, 9(6):717–772, 2009.

[4] S. Chang, G.-J. Qi, J. Tang, Q. Tian, Y. Rui, and T. S. Huang. Multimedia lego: Learning structured model by probabilistic logic ontology tree. In *ICDM*, pages 979–984. IEEE, 2013.

[5] J. M. Engels and P. Diehr. Imputation of missing longitudinal data: a comparison of methods. *Journal of clinical epidemiology*, 56(10):968–976, 2003.

[6] G. C. Fonarow, K. F. Adams Jr, W. T. Abraham, C. W. Yancy, W. J. Boscardin, et al. Risk stratification for in-hospital mortality in acutely decompensated heart failure. *JAMA*, 293(5):572–580, 2005.

[7] P. Gong, J. Ye, and C. Zhang. Multi-stage multi-task feature learning. In *NIPS*, pages 1997–2005, 2012.

[8] P. Gong, J. Ye, and C. Zhang. Robust multi-task feature learning. In *KDD*, pages 895–903. ACM, 2012.

[9] T. Hastie, R. Tibshirani, G. Sherlock, M. Eisen, P. Brown, and D. Botstein. Imputing missing data for gene expression arrays. *Tech. Report*, 1999.

[10] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *JMLR*, 5:1457–1469, 2004.

[11] P. Jain, R. Meka, and I. Dhillon. Guaranteed rank minimization via singular value projection. In *NIPS*, pages 937–945. 2010.

[12] Y. Koren. Collaborative filtering with temporal dynamics. In *KDD*, pages 447–456, 2009.

[13] T. A. Lasko, J. C. Denny, and M. A. Levy. Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data. *PloS one*, 8(6):e66341, 2013.

[14] J. Lee, Y. Sun, and M. Saunders. Proximal newton-type methods for convex optimization. In *NIPS*, volume 25, pages 836–844. 2012.

[15] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections*. Arizona State University, 2009.

[16] Z. Lu, D. Agarwal, and I. S. Dhillon. A spatio-temporal approach to collaborative filtering. In *Proc. of the 3rd ACM Conf. on Rec. Sys.*, pages 13–20, 2009.

[17] M. Markatou, P. K. Don, J. Hu, F. Wang, J. Sun, R. Sorrentino, and S. Ebadollahi. Case-based reasoning in comparative effectiveness research. *IBM J. of Res. and Dev.*, (5):4, 2012.

[18] R. Mazumder, T. Hastie, and R. Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *The J. of Mach. Learn. Res.*, 99:2287–2322, 2010.

[19] S. D. Persell, A. P. Dunne, D. M. Lloyd-Jones, and D. W. Baker. Electronic health record-based cardiac risk assessment and identification of unmet preventive needs. *Med. Care*, 47(4):418–424, 2009.

[20] E. F. Philbin and T. G. DiSalvo. Prediction of hospital readmission for heart failure: development of a simple risk score based on administrative data. *J. of the Am. Co. of Card.*, 33(6):1560–1566, 1999.

[21] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. *NIPS*, 20:1257–1264, 2008.

[22] T. Schneider. Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *J. of Climate*, 14(5):853–871, 2001.

[23] Y. Shen, Z. Wen, and Y. Zhang. Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization. *Opti. Met. and Soft.*, 0(0):1–25, 2012.

[24] M. Stern, K. Williams, D. Eddy, and R. Kahn. Validation of prediction of diabetes by the archimedes model and comparison with other predicting models. *Diab. Care*, 31(8):1670–1671, 2008.

[25] J. Sun, F. Wang, J. Hu, and S. Edabollahi. Supervised patient similarity measure of heterogeneous patient records. *SIGKDD Explorations*, (1):16–24, 2012.

[26] T. Van Staa, H. Leufkens, and C. Cooper. Utility of medical and drug history in fracture risk prediction among men and women. *Bone*, 31(4):508–514, 2002.

[27] F. Wang, N. Lee, J. Hu, J. Sun, and S. Ebadollahi. Towards heterogeneous temporal clinical event pattern discovery: a convolutional approach. In *KDD*, pages 453–461, 2012.

[28] Z. Wen, W. Yin, and Y. Zhang. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Math. Prog. Comp.*, 4(4):333–361, 2012.

[29] S. J. Wright, R. D. Nowak, and M. A. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Trans. on Sig. Proc.*, 57(7):2479–2493, 2009.

[30] J. Wu, J. Roy, and W. F. Stewart. Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches. *Med. Care*, 48(6):S106, 2010.

[31] L. Xiong, X. Chen, T.-K. Huang, J. Schneider, and J. G. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *SDM*, 2010.

[32] J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. In *NIPS*, pages 702–710. 2011.

[33] J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. Arizona State University, 2011. `http://www.MALSAR.org`.

[34] J. Zhou, J. Liu, V. A. Narayan, and J. Ye. Modeling disease progression via multi-task learning. *NeuroImage*, 78:233–248, 2013.

[35] J. Zhou, Z. Lu, J. Sun, L. Yuan, F. Wang, and J. Ye. Feafiner: biomarker identification from medical data through feature generalization and selection. In *KDD*, pages 1034–1042. ACM, 2013.

[36] J. Zhou, J. Sun, Y. Liu, J. Hu, and J. Ye. Patient risk prediction model via top-k stability selection. In *SDM*. SIAM, 2012.

[37] J. Zhou, L. Yuan, J. Liu, and J. Ye. A multi-task learning formulation for predicting disease progression. In *KDD*, pages 814–822. ACM, 2011.