Granite: Diversified, Sparse Tensor Factorization for Electronic Health Record-Based Phenotyping

Jette Henderson*, Joyce C. Ho†, Abel N. Kho‡, Joshua C. Denny§, Bradley A. Malin§, Jimeng Sun¶, Joydeep Ghosh*
* University of Texas at Austin, † Emory University, ‡ Northwestern University
§ Vanderbilt University, ¶Georgia Institute of Technology

***Abstract*—One of the most formidable challenges electronic health records (EHRs) pose for traditional analytics is the inability to map directly (or reliably) to medical concepts or phenotypes. Among other things, EHR-based phenotyping can help identify and target patients for interventions and improve real-time clinical decisions. Existing phenotyping approaches often require labor-intensive supervision from medical experts or do not focus on generating concise and diverse phenotypes. Sparsity in phenotypes is key to making them interpretable and useful to clinicians, while diversity allows clinicians to grasp the main features of a patient population quickly.**

In this paper, we introduce Granite, a diversified, sparse nonnegative tensor factorization method to derive phenotypes with limited human supervision. Compared to existing high-throughput phenotyping techniques, Granite yields phenotypes with much more distinct (non-overlapping) elements that can, as an artifact, capture rare phenotypes. Moreover, the resulting concise phenotypes retain predictive powers comparable to or surpassing existing dimensionality reduction techniques. We evaluate Granite by comparing its resulting phenotypes with those generated using state-of-the-art, high-throughput methods on simulated as well as real EHR data. Our algorithm offers a promising and novel data-driven solution to rapidly characterize, predict, and manage a wide range of diseases.

***Keywords*—Feature extraction; Data mining; Health information management; Computational phenotyping; Tensor factorization; Electronic health records**

I. Introduction

Computational phenotyping is the process of extracting clinically relevant and interesting characteristics from a set of clinical documentation, such as that which is recorded in electronic health records (EHRs). Computational phenotyping can be viewed as a form of dimensionality reduction, where each phenotype forms a latent space [1]. Currently, there are two approaches to deriving computational phenotypes, which are: 1) rule-based methods derived by domain experts and 2) automatic high-throughput methods, using machine learning and data mining. In the first approach, panels of experts define a single phenotype by a series of rules (see [2] for an example of this method). While they are based on consensus of knowledgeable individuals, rule-based methods are limited in that they are laborious, iterative, and time-consuming processes [3].

The second approach, which has gained traction in the past few years [4] and is the focus of this work, automatically extracts phenotypes in a high-throughput manner via machine learning and data mining methods. Nonnegative tensor factorization (NNTF) on tensors constructed from EHR data is one way to perform high-throughput phenotyping. Tensors are multidimensional arrays (an example of a tensor can be seen in Figure 1), and tensor factorization utilizes multiway structure and relationships to produce results in an unsupervised manner that are potentially more interpretable than other methods. Using NNTF to perform high-throughput phenotyping was initially proposed through a method called *Limestone*, which showed that NNTF could computationally extract candidate phenotypes, a surprisingly large number of which were deemed clinically relevant by medical experts [5]. However, one of Limestone's drawbacks is that it relies upon post-processing to create more sparsity in the phenotypes. A subsequent algorithm called *Marble* addressed this weakness in Limestone by directly adding a global offset tensor and employing a new inference method to encourage sparsity and stability in the phenotypes [6]. However, the phenotypes themselves contain less diversity (i.e., more overlap, see Figure 3b), which makes them less useful to clinicians. Additionally, Marble's sparsity parameter must be set by the user manually rather than fit.

High-throughput phenotyping has also been achieved with other machine learning and data mining techniques. [7] had success applying weakly supervised matrix factorization to clinical notes to generate phenotypes when the conditions were known a priori, while others have used matrix factorization on the micro (patient) and macro (population) to derive sparse phenotypes from longitudinal EHR data [8]. Other methods have delivered insights using topic modeling approaches. Some topic modeling methods focused solely on diagnosis codes [9] and others on heterogenous data (e.g., diagnosis, laboratory results, clinical notes) [10]. Further investigations have applied deep learning to raw EHR data with success, but their methods require supervision, which is not always available in data sources or may be too restrictive for a phenotyping task [11]–[13]. While the work above delivers insight into patient populations, only [8] focuses on creating concise phenotypes, and none of the above generate diverse phenotypes. For clinicians, diversity is important to discover rare phenotypes in a patient population as well as in features in predictive models. Moreover, diverse phenotypes are likely easier to implement, as a clinician may find it difficult to rank-order or apply phenotypes that have substantial overlap.

To answer this need, we introduce *Granite*, a novel NNTF model to fit count data, that produces diverse, sparse, and interpretable candidate phenotypes in an unsupervised manner. Granite deviates from Marble [6] in several key aspects: (i) it introduces a flexible penalized angular regularization term on the factors to promote diversity, (ii) it utilizes a simplex

978-1-5090-4881-6/17 $31.00 © 2017 IEEE
DOI 10.1109/ICHI.2017.61

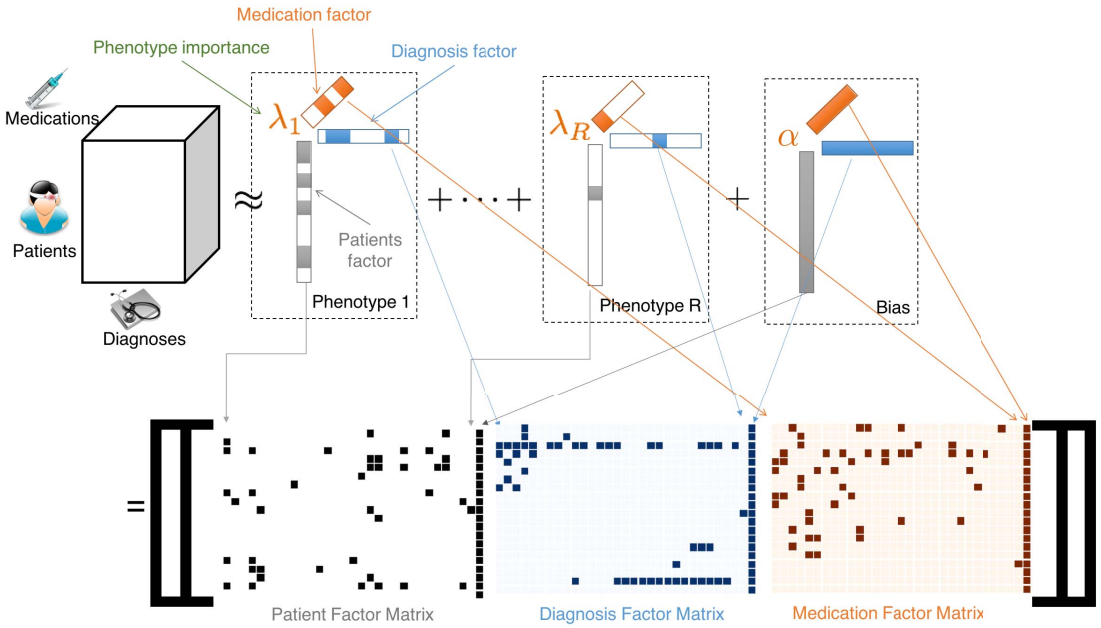


Fig. 1: Overview of phenotyping via tensor decomposition process. A tensor is constructed of patient-level data is decomposed into the weighted sum of rank-one tensors based on the minimization of an objective function. Each rank-one tensor, formed by taking the outer product of factor vectors, constitutes a phenotype.

projection to calculate the factors and ℓ_2-regularization to achieve better sparsity control, and (iii) it develops an effective projected gradient descent-based approach to solve for the interaction and bias factors simultaneously. The penalized angular regularization term is flexible so users can encode different amounts of diversity in each mode. We illustrate the efficacy of our model on simulated data and real EHR data.

II. Preliminaries and Related Work

This section introduces the preliminaries of matrix and tensor decomposition and related phenotyping via tensor factorization work. For the purposes of indexing a matrix $\mathbf{A}$, we denote the r^{th} column as $\mathbf{a}_r$. The definition for the algebraic operations used in the paper are provided below.

Definition 1: The Khatri-Rao product of two matrices $\mathbf{A} \odot \mathbf{B}$ of sizes $I_A \times R$ and $I_B \times R$ respectively, produces a matrix $\mathbf{Z}$ of size $I_A I_B \times R$ such that $Z = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \cdots \quad \mathbf{a}_R \otimes \mathbf{b}_R]$, where $\otimes$ represents the Kronecker product. The Kronecker product of two vectors $\mathbf{a} \otimes \mathbf{b} = [a_1\mathbf{b} \quad a_2\mathbf{b} \cdots a_{I_A}\mathbf{b}]^T$

Definition 2: The element-wise multiplication (and division) of two same-sized matrices $\mathbf{A} * \mathbf{B}$ ($\mathbf{A} \oslash \mathbf{B}$) produces a matrix $\mathbf{Z}$ of the same size such that the element $c_{\vec{i}} = a_{\vec{i}} b_{\vec{i}}$ ($c_{\vec{i}} = a_{\vec{i}} / b_{\vec{i}}$) for all $\vec{i}$.

Definition 3: $\boldsymbol{\mathcal{X}}$ is an N-way rank one tensor if it can be written as the outer product of N vectors, $\mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \cdots \circ \mathbf{a}^{(N)}$, where each element $x_{\vec{i}} = x_{i_1, i_2, \cdots, i_N} = a^{(1)}_{i_1} a^{(2)}_{i_2} \cdots a^{(N)}_{i_N}$.

A. Tensor Factorization

A tensor is a generalization of a matrix to a multidimensional array. Each element of a tensor represents an n-way interaction (e.g., a third order tensor could capture the relationship between a document, term, and author). Tensors can be decomposed into a product of matrices or a combination of matrices and smaller tensors. Tensor factorization utilizes information in the multiway structure to produce factors that are concise, potentially more interpretable (than matrix methods), even with relatively small amounts of observations [14].

Many tensor decomposition models exist and a complete review of all the techniques is beyond the scope of the paper. Instead, we focus on the CANDECOMP/PARAFAC (CP) decomposition [15], [16], a common tensor factorization model. CP decomposition factorizes the original tensor $\boldsymbol{\mathcal{X}}$ as a sum of R rank-one tensors and can be expressed as follows (see Figure 1):

$$\boldsymbol{\mathcal{X}} \approx \sum_{r=1}^{R} \lambda_r \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(N)} = [\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \ldots; \mathbf{A}^{(N)}]\!] \quad (1)$$

The latter representation is shorthand notation for the weight vector $\boldsymbol{\lambda} = [\lambda_1 \cdots \lambda_R]$ and the factor matrix $\mathbf{A}^{(n)} = [\mathbf{a}_1^{(n)} \cdots \mathbf{a}_R^{(n)}]$. Standard CP decomposition is formulated as a least squares approximation, called CP alternating least squares (CP-ALS), where data is assumed to follow a Gaussian distribution, which makes it well-suited for continuous data [14]. This assumption also results in simpler algorithms, and the Alternating Direction Method of Multipliers (ADMM) technique can be readily applied for distributed computation. However, since the kind of EHR data considered in this work is based on counts, a better match is the nonnegative CP alternating Poisson regression (CP-APR) model developed in [17], wherein the objective is to minimize the KL divergence (i.e., data follows Poisson distribution).

B. Phenotyping via Tensor Factorization

Figure 1 shows an example of the phenotyping process using CP decomposition. The input to the model is a tensor composed of three modes, patients, their diagnoses, and their medications. The output is a weighted sum of rank-one tensors. Each rank-one tensor is formed by taking the outer product of three factor vectors that are found by solving an optimization problem for each of the three modes. These factor vectors can be organized into factor matrices by mode, which is depicted in the lower part of Figure 1 (note: the weights, $\boldsymbol{\lambda}$, have been absorbed into the patient factor matrix). Factor matrices are a convenient way to keep track of the modes in a decomposition.

Several NNTF models have been proposed to achieve high-throughput computational phenotyping with minimal human intervention [5], [6], [18]–[20]. Limestone obtains phenotypes by decomposing the EHR tensor using the CP-APR algorithm and post-processing the factors to remove probabilistically unlikely elements [5]. Marble uses a bias tensor and a user-specified sparsity threshold to produce sparse factors. The factor matrices in Figure 1 come from a Marble decomposition of a patient $\times$ diagnosis $\times$ medication tensor (the first five phenotypes of this fit are shown in Figure 3). Summing across columns gives the number of phenotypes that contain a particular diagnosis, medication, or patient. For example, the third row of the diagnosis factor matrix is "Major Symptoms, Abnormalities," which appears in the majority of the phenotypes. Domain experts were critical of the fact that while Marble produces interpretable, concise phenotypes, there was too much similarity across phenotypes. Granite addresses this weakness through an angular penalty to increase intra-phenotype diversity.

Other tensor factorization methods have been proposed. Taking a different approach, [18] introduced a sparse Hierarchical Tucker Factorization, which uses a network of tensors. The authors showed how it could be applied to extracting diagnosis phenotypes out of EHR data using the hierarchical structure in ICD-9 codes. While Granite does not incorporate the hierarchy of the EHR record, it does capture the multiway interaction between different types of patient interactions with the medical system (e.g., diagnosis and medication). [19] imposes pairwise constraints on the vectors in the factor matrices, but these constraints result in solutions with near orthogonal vectors. While this approach provides high-level insights into a patient population, it may smooth over more nuanced medical realities. Granite encourages sparse and diverse phenotypes, and it has the potential to isolate small, rare phenotypes. [20] used a Bayesian NNTF approach to decompose an EHR count tensor. However, unlike Granite, this model does not induce sparsity and diversity in those phenotypes.

III. Granite: Diversity-Promoting Tensor Factorization

Granite is a robust Poisson NNTF model that encourages diverse and sparse latent factors. There are two main differences between Granite and Marble: (1) the introduction of an angular penalty term and an ℓ_2 regularization term on the signal factors substantially reduces overlaps between the factors and (2) simplex projection on the factors, which, as we will empirically demonstrate, results in better sparsity control.

A. Problem Formulation

Let $\boldsymbol{\mathcal{X}}$ denote an $I_1 \times I_2 \times \cdots \times I_N$ tensor of count (nonnegative integer) data and $\boldsymbol{\mathcal{Z}}$ represent a same-sized tensor where each element $z_{\vec{i}}$ contains the optimal Poisson parameters of the observed tensor $x_{\vec{i}}$. The Granite optimization problem is defined as the following:

$$\min(f(\boldsymbol{\mathcal{X}})) \equiv \min(\sum_{\vec{i}}(z_{\vec{i}} - x_{\vec{i}} \log z_{\vec{i}}) \tag{2}$$

$$+\underbrace{\frac{\beta_1}{2}\sum_{n=1}^{N}\sum_{r=1}^{R}\sum_{p=1}^{r}(\max\{0, \frac{(\mathbf{a}_p^{(n)})^\intercal \mathbf{a}_r^{(n)}}{||\mathbf{a}_p^{(n)}||_2||\mathbf{a}_r^{(n)}||_2} - \theta_n\})^2}_{\text{angular regularization}} \tag{3}$$

$$+\underbrace{\frac{\beta_2}{2}\sum_{n=1}^{N}\sum_{r=1}^{R}||\mathbf{a}_r^{(n)}||_2^2)}_{\ell_2\text{regularization}} \tag{4}$$

$$\text{s.t } \boldsymbol{\mathcal{Z}} = [\![\sigma; \mathbf{u}^{(1)}; \cdots; \mathbf{u}^{(N)}]\!] + [\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}]\!] \tag{5}$$

$$\sigma > 0, \lambda_r \geq 0, \ \forall r$$

$$\mathbf{A}^{(n)} \in [0,1]^{I_n \times R}, \mathbf{u}^{(n)} \in (0,1]^{I_n \times 1}, \ \forall n$$

$$||\mathbf{a}_r^{(n)}||_1 = ||\mathbf{u}^{(n)}||_1 = 1, \ \forall n. \tag{6}$$

Minimizing the objective function, f, in Equations (2, 3, 4) results in the tensor $\boldsymbol{\mathcal{Z}}$. As shown in Equation (5), $\boldsymbol{\mathcal{Z}}$ consists of two terms: (i) rank-one bias tensor with positive weight and factor vectors, σ and $\mathbf{u}^{(1)}, \cdots, \mathbf{u}^{(1)}$, and (ii) rank R interaction tensor with nonnegative weight vector and factor matrices, $\boldsymbol{\lambda}$ and $\mathbf{A}^{(1)}, \cdots, \mathbf{A}^{(N)}$. The rank R interaction tensor is composed of the weighted sum of rank-one tensors. Each rank-one tensor is constructed from N stochastic vectors (elements sum to 1 and are nonnegative), which is consistent with the existing CP Poisson tensor decompositions. We now discuss key features of the Granite approach in more detail.

1) Promoting Intra-Phenotype Diversity: To encourage diversity between the rank-one tensors, Granite introduces a penalty term to the objective function, shown in Equation (3). The penalized angular regularization term reduces the occurrence of overlapping elements in the interaction factor matrices $\mathbf{A}^{(n)}$ by penalizing decompositions where the factor vectors are too correlated, measured by the cosine of the angle between the vectors. Two vectors that are orthogonal will yield a cosine similarity of 0, while two identical vectors will result in a 1. This penalty is adapted from [21], which introduced angular constraints to yield a structure-revealing data fusion model that is robust to overfactoring. However, our model relaxes the angular constraint and softly imposes diversity via the regularization penalty. This results in the flexibility to allow for overlapping phenotypes in the scenario where it truly exists.

It is also important to note that *only vectors whose cosine angle with other vectors are greater than θ_n are penalized*. Thus, our model does not necessarily encourage orthogonal factor components unless $\theta_n = 0$, which would result in similar constraints as in [19]. Since θ_n is specific to each mode, our model can impose different levels of diversity on each mode. A user may want to focus on extracting a few, diverse diagnoses but be less concerned with the similarity between the vectors of the patient mode.

2) Promoting Inter-Phenotype Sparsity: Granite uses ℓ_2-regularization (see Equation(4)) and simplex projection (see

Section III-B2) to achieve sparsity. Experimentally, ℓ_2-regularization term encourages the terms in the factor matrix vectors to be small. In Granite, the terms are projected back into feasible space using simplex projection onto a ball of diameter s and then are ℓ_1 renormalized. Adjusting the size of parameter s determines the number of non-zero terms in the factor vectors. The ℓ_2-regularization term along with the simplex projection act like an Elastic Net [22] regularization to drive terms in the interaction tensors to 0.

3) Capturing the Baseline: The bias tensor, carried over from the Marble framework, captures the general features of the tensor and provides the stability necessary for elements in the factor vectors to be driven to zero. The bias tensor encapsulates the general characteristics of a patient population while the R rank-one interaction tensors reflect the key features of subgroups of the patient population.

B. Algorithm

1) Gradient: The Granite algorithm minimizes the objective function f to solve for the bias and interaction factor matrices simultaneously through projected gradient descent. The approach is different than Marble. Specifically, Marble combines an alternating minimization approach, where each mode has a multiplicative update with a sequential unconstrained minimization approach. Not only have gradient descent approaches been shown to have faster convergence compared to the alternating minimization approach [23], but the projected gradient step avoids the problem of zeroing out components too early in the multiplicative updates. Furthermore, solving for the bias and interaction terms simultaneously avoids a potential problem where subtracting the best rank-one approximation may actually increase the tensor rank [24]. We note that although recent work [25] obtained better speed and accuracy of CP decomposition of Poisson data using bound-constrained Newton methods, the angular regularization term results in complications for second-order optimization.

Granite combines the interaction and bias vectors for each factor matrix, such that for mode n, the combined factor matrix is $\hat{\mathbf{A}}^{(n)} = \begin{bmatrix}\mathbf{A}^{(n)} & \mathbf{u}^{(n)}\end{bmatrix}$. Our preliminary experiments, which are not shown due to space constraints, showed that absorbing the weights, $\boldsymbol{\lambda}$ and σ, into one of the modes cut down on computation time as well as increased the stability of the results. Without loss of generality, the first mode is chosen to be $\hat{\mathbf{A}}^{(1)} = \begin{bmatrix}\left(\lambda\mathbf{A}^{(1)}\right) & (\sigma\mathbf{u}^{(1)})\end{bmatrix}$.

Before we show the partial derivative for the factor vector $\mathbf{a}_r^{(n)}$, we introduce some notational conveniences. The objective function, f, can be represented as a scalar-valued function of the parameter vector $\mathbf{y}$ [23], where $\mathbf{y}$ represents either the vectorization of the factor matrices or the weights.

$$\mathbf{y} = \begin{bmatrix} \text{vec}(\lambda\mathbf{A}^{(1)}) & \sigma\mathbf{u}^{(1)}) \\ \text{vec}(\mathbf{A}^{(2)} & \mathbf{u}^{(2)}) \\ \vdots & \\ \text{vec}(\mathbf{A}^{(N)} & \mathbf{u}^{(N)}) \end{bmatrix} = \begin{bmatrix} \text{vec}(\hat{\mathbf{A}}^{(1)}) \\ \text{vec}(\hat{\mathbf{A}}^{(2)}) \\ \cdots \\ \text{vec}(\hat{\mathbf{A}}^{(N)}) \end{bmatrix}$$

Then, the gradients of the objective function f can be formed by vectorizing the partial derivatives with respect to each component of the parameter vector $\mathbf{y}$:

$$\nabla f(\mathbf{y}) = \begin{bmatrix}\text{vec}\left(\frac{\partial f}{\partial \hat{\mathbf{A}}^{(1)}}\right) & \cdots & \text{vec}\left(\frac{\partial f}{\partial \hat{\mathbf{A}}^{(N)}}\right)\end{bmatrix}^\intercal$$

For notation purposes, we can represent the matricized form of the tensor decomposition as:

$$[\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}]\!]_{(n)} = \boldsymbol{\lambda}\mathbf{A}^{(n)}(\mathbf{A}^{(-n)})^\intercal$$

where

$$\mathbf{A}^{(-n)} \equiv \mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)}$$

The partial derivatives with respect to the factor matrices are the following:

$$\begin{aligned}\frac{\partial f}{\partial \mathbf{a}_r^{(n)}} &= [1 - \mathbf{X}_{(n)} \oslash \mathbf{Z}_{(n)}]\mathbf{a}_r^{(-n)} + \beta_2\ \mathbf{a}_r^{(n)} \\ &\quad + \beta_1 \sum_{p \neq r} \left(\max\{0, g(\mathbf{a}_r^{(n)}, \mathbf{a}_p^{(n)})\}\right) \frac{\partial g(\mathbf{a}_r^{(n)}, \mathbf{a}_p^{(n)})}{\partial \mathbf{a}_r^{(n)}}\end{aligned} \tag{7}$$

$$\frac{\partial f}{\partial \mathbf{u}^{(n)}} = [1 - \mathbf{X}_{(n)} \oslash \mathbf{Z}_{(n)}]\mathbf{u}^{(-n)} \tag{8}$$

Further details about the derivation of the gradient are in the appendix.

2) Projection: Projected gradient descent is used to ensure the solution lies in the feasible space (i.e., non-negative or positive). For the first mode, $\mathbf{A}^{(1)}$ and $\mathbf{u}^{(1)}$, the projection function is simply the standard projection on the nonnegative and positive orthant respectively:

$$P_{\mathbf{A}}(\mathbf{A}^{(1)}) = \max\{0, \mathbf{a}_r^{(1)}\}, \tag{9}$$

$$P_{\mathbf{u}}(\mathbf{u}^{(1)}) = \max\{\epsilon, \mathbf{u}^{(1)}\}, \epsilon \text{ positive \& infinitesimal.} \tag{10}$$

Projection of the other bias vectors for the other modes occurs identically to Equation 10.

Projection for the interaction factor components $\mathbf{a}_r^{(g)}$ other than the first mode uses the Euclidean projection onto the ℓ_1-ball of diameter s [26]. When $s = 1$, this is projection onto the probabilistic (or canonical) simplex. However, Granite takes advantage of the properties of the simplex projection and decreases s to a number less than 1, which results in even more sparse solutions. The subsequent result is then renormalized to meet the stochastic constraints. The detailed Granite algorithm is presented in Algorithm 1, with further details are located in the appendix.

3) Membership of Existing Factors: Granite also computes a membership vector for a new axis, where the other modes are fixed with the already learned factors. The membership vector is defined as the convex combination of existing tensor factors, where the r^{th} element denotes the probability the entity exhibits characteristics consistent with the r^{th} tensor factors. For example, new patients can be projected onto the computational phenotypes to obtain a *phenotype membership* vector where each element represents the probability the patient has the phenotype. It is important to note that the membership vector is not equivalent to the factor matrix because the *stochastic constraints are on the row* and not the column. The ability to take new patients and obtain their phenotype membership can be used in several ways. For one, predictive models can be trained on phenotypes associated with a subset of population and then applied to other subsets.

Algorithm 1: Granite algorithm

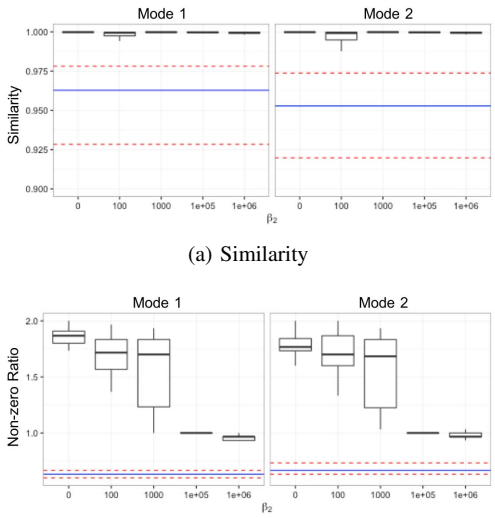
Data: $\boldsymbol{\mathcal{X}}, R, \mathbf{s}, \boldsymbol{\theta}$
Result: $[\![\sigma; \mathbf{u}^{(1)}; \cdots; \mathbf{u}^{(N)}]\!], [\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}]\!]$
for $k = 1, 2, \cdots, K$ **do**
  `# Update parameters` $\hat{\mathbf{A}}^{(n)}$
  Calculate $\nabla f(\hat{\mathbf{A}}^{(n)})$ for $n = 2, \ldots, N$ using Eqs. (7, 8)
  `#Simplex projection with` $\mathbf{s}$
  Update $\hat{\mathbf{A}}^{(n)}$ for $n = 2, \ldots, N$ with projected gradient descent line search and simplex projection
  `# Update parameter` $\hat{\mathbf{A}}^{(1)}$
  Calculate $\nabla f(\hat{\mathbf{A}}^{(1)})$ using Eqs. (7, 8)
  Update $\hat{\mathbf{A}}^{(1)}$ with gradient descent and nonnegative projection Eqs. (9, 10)
  `# Standard stopping criteria`
  if $||\mathbf{y}^{+} - \mathbf{y}||_F <$ convergenceTol **then**
    break
  end
end

Without loss of generality, we assume the 1st mode is the new axis (e.g., patients). Given a new tensor $\tilde{\boldsymbol{\mathcal{X}}}$, we want to find $\tilde{\mathbf{A}}^{(1)}, \tilde{\mathbf{u}}^{(1)}$ that provides the best approximation given $\mathbf{A}^{(2)}, \cdots, \mathbf{A}^{(N)}$ are fixed. We observe that this is almost equivalent to gradient descent where the partial derivatives of the other factors are zero except that the membership vector is obtained by normalizing the entries of $\tilde{\mathbf{A}}^{(1)}$ across the row instead of the columns. To solve for the optimal $\tilde{\mathbf{A}}^{(1)}, \tilde{\mathbf{u}}^{(1)}$, the same projected gradient descent approach described in Section III-B2 is taken with the projection onto the nonnegative orthant and the angular and ℓ_2 regularization penalties set to zero (minimizing the KL divergence only). Once $\tilde{\mathbf{A}}^{(1)}$ is calculated, the rows are normalized to sum to 1. A detailed algorithm is provided in the appendix.

IV. Simulation Results

In this section, we evaluate Granite's performance on simulated tensors where the actual factors are known, which allows us to demonstrate the recovery properties of Granite in a controlled environment and explore the effects of algorithmic choices. Specifically, we consider a third-order tensor of size $40 \times 20 \times 20$ with rank of 5 (i.e., $R = 5$). We generate the model $\boldsymbol{\mathcal{Z}} = [\![\sigma; \mathbf{u}^{(1)}; \cdots; \mathbf{u}^{(N)}]\!] + [\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}]\!]$. Both the weights and bias factor vectors are straightforward, as the sampling occurs in the nonnegative and positive orthants respectively. We simulate the vectors in each interaction factor matrix $\mathbf{A}^{(n)}$ by sampling non-zero element indices according to a specified sparsity pattern. We then randomly sample along the simplex for the non-zero indices, rejecting vectors that are too similar to those already generated. Finally, each tensor element x_{ijk} is sampled from the Poisson distribution with the parameter set to z_{ijk}.

Our algorithm is evaluated on 50 simulated tensors where we set the cosine similarity to .3 and $\beta_1 = 1000$ and varied β_2 for each run. In addition, we fixed the sparsity parameter to project onto the simplex ($s = 1$) for the first mode and $s = .95$ for the second and third modes. The results are evaluated using



(a) Similarity

(b) Non-zero Ratio

Fig. 2: Similarity (top) and non-zero ratio (bottom) between the fit latent factors, calculated by Granite and Marble, and the true latent factors for the second and third mode. The boxes represent Granite's performance, and the median and the the 25% and 75% percentiles of Marble's performance are designated by the blue and red dotted lines, respectively.

1) the non-zero ratio between the computed solution and the actual solution and 2) the cosine angle between vectors in the simulated and fit tensors. The cosine angle between the two vectors, a component of the factor match score [17], is used to quantify the similarity between the computed solution and the actual representation. We use the Hungarian method to compute the optimal pairing between each approximated rank-one tensor and the corresponding "true" rank-one representation.

Figure 2a shows a boxplot of the similarity scores between the estimated latent factors and the true latent factors for the second and third mode, where the blue line represents the median performance of Marble and the red dotted lines are the 25% and 75% percentiles of Marble's similarity scores. Overall, Granite is able to recover the true latent representation with similarity scores near 1, which surpasses the median similarity scores of Marble.Figure 2b illustrates the non-zero ratio (i.e., (number of non-zeros in fitted factor vectors)/(number of non-zeros in actual factor vectors)) with the blue line denoting the median non-zero ratio of Marble. Granite's non-zero ratio improves as β_2 increases and the algorithm is able to recover the original sparsity pattern. While Marble's non-zero ratio is lower overall, it is below the original sparsity pattern and Granite outperforms Marble in terms of recovering the original tensor. Thus, Granite is able to capture the simulated latent factors while maintaining sparse solutions.

V. Empirical Results

A. Dataset Description

The Synthetic Derivative (SD) is a large, de-identified Electronic Medical Record (EMR) database at the Vanderbilt University Medical Center (VUMC) [27]. Among other pieces

Phenotype 1
(15.43% of Patients)
Legally Blind
Major Symptoms, Abnormalities (1,2)
Polyneuropathy
Cerebrovascular Disease Late Effects, Unspecified
Multiple Sclerosis
anticonvulsants
bronchodilators
anxiolytics, sedatives, and hypnotics

Phenotype 2
(10.76% of Patients)
Specified Heart Arrhythmias
Major Symptoms, Abnormalities (1,2)
Heart Infection/Inflammation, Except Rheumatic
diuretics
beta-adrenergic blocking agents
antihyperlipidemic agents (2,5)

Phenotype 3
(5.92% of Patients)
Other Endocrine/Metabolic/Nutritional Disorders (3,5)
Severe Hematological Disorders
vitamins

Phenotype 4
(3.41% of Patients)
Rheumatoid Arthritis and Inflammatory Connective Tissue Disease
antirheumatics

Phenotype 5
(7.71% of Patients)
Other Endocrine/Metabolic/Nutritional Disorders (3,5)
antihyperlipidemic agents (2,5)

(a) The top 5 Granite phenotypes.

Phenotype 1
(13.27% of Patients)
Other Infectious Diseases (1,2,5)
Bone/Joint/Muscle Infections/Necrosis (ii)
Major Symptoms, Abnormalities (1,2,3,4,5)
antiemetic/antivertigo agents (1,2)
anticonvulsants
anxiolytics, sedatives, and hypnotics
antihistamines (1,2)

Phenotype 2
(9.6% of Patients)
Severe Hematological Disorders
Major Symptoms, Abnormalities (1,2,3,4,5)
Parkinson's and Huntington's Diseases
analgesics
antiemetic/antivertigo agents (1,2)
antihistamines (1,2)

Phenotype 3
(5.38% of Patients)
Other Infectious Diseases (1,2,5)
Bone/Joint/Muscle Infections/Necrosis (ii)
Major Symptoms, Abnormalities (1,2,3,4,5)
antifungals
antituberculosis agents
dermatological agents

Phenotype 4
(15.43% of Patients)
Major Symptoms, Abnormalities (1,2,3,4,5)
Coronary Atherosclerosis/Other Chronic Ischemic Heart Disease
Congestive Heart Failure
Hypertension
beta-adrenergic blocking agents
diuretics
antiarrhythmic agents
antihyperlipidemic agents

Phenotype 5
(5.38% of Patients)
Major Symptoms, Abnormalities (1,2,3,4,5)
Other Infectious Diseases (1,2,5)
laxatives
antacids
mouth and throat products
antiseptic and germicides

(b) The top 5 Marble phenotypes.

Fig. 3: The Granite and Marble phenotypes with the highest weights (i.e., largest λ_is) for parameters listed in Table I. Capitalized blue items are diagnoses, and lower case orange items are medications. Numbers indicate phenotypes where item occurs.

of patient information, the SD contains inpatient and outpatient billing and medication codes of nearly 2 million patients. In the work of [28], domain experts manually developed algorithms that use these codes to identify case and control statuses for patients within the SD for certain conditions.

In this work, we focus on the patients identified as case and controls for resistant hypertension. For each patient in the tensor, we include five years of data from the last diagnosis they received. We construct the count tensor from medication and diagnosis records. Since individual International Classification of Diseases (ICD-9) diagnosis codes capture information at a fine-grained level specialized for billing purposes, we use CMS's Hierarchical Condition Categories (HCC) to group the diagnosis codes.[1] Additionally, we aggregate medications based on Medical Subject Headings (MeSH) pharmacological actions provided by the RxClass RESTful API, a product of the US National Library of Medicine.[2] It is important to note that a medication may have several uses and, therefore, belong to multiple categories. The resulting tensor is 1394 patients by 149 medications by 177 diagnosis and thus has over 36 million cells. Of these patients, 33% of the patients were labeled as resistant hypertension cases and 67% were labeled as controls.

B. Results

We evaluate Granite against other dimensionality reduction techniques in the following three ways: 1) we quantitatively compare Granite-generated phenotypes with Marble-generated phenotypes to demonstrate the desirable qualities of Granite, 2) we use annotations from a domain expert to analyze the clinical relevance of the phenotypes, and 3) we use the phenotypes generated in an unsupervised manner as features in a supervised classification task to demonstrate the predictive power of Granite.

[1]http://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp
[2]https://rxnav.nlm.nih.gov/RxClassAPIs.html

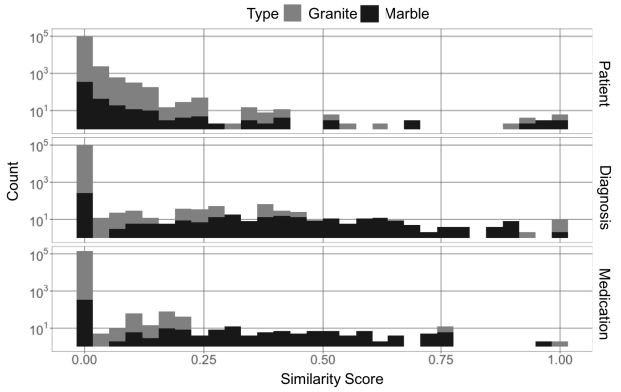


Fig. 4: Cosine similarity within factor matrices for Granite and Marble, run with parameter values listed in Table I (counts are shown on a log scale).

First, we compare phenotypes generated with Granite with those generated by Marble. Using a grid search, we chose the parameter values listed in Table I for all described results. Figure 3 shows the Granite- (top) and Marble-generated phenotypes (bottom) associated with the largest weights, $\boldsymbol{\lambda}$. The numbers in parentheses next to the items indicate in which phenotypes the items appear. For example, "Other infectious diseases" is labelled "(1, 3, 5)" because it is repeated in Phenotypes 1, 3, and 5 in the Marble-generated phenotypes. Overall, Granite produces more diverse phenotypes, which is illustrated in Figure 4. Using a log scale for the counts,

TABLE I: Values of experimental parameters.

Method	Parameters
Granite	$R = 30, s = [1, .99, .99], \theta = [1, .35, .35], \beta_1 = 10000, \beta_2 = 1000$
Marble	$R = 30, \alpha = 10000, \gamma = [0, .15, .15]$

Figure 4 shows histograms of the cosine similarity scores between vectors by mode. Here, the angular penalty term for the Granite decomposition was set to .35, and in the histogram, it can be seen that the vectors in the Granite factor matrices have cosine scores between 0 (completely perpendicular) and .4 (a small number of common terms) in the diagnosis mode and 0 and .25 in the medication mode. Note that, since the angular penalty was set to 1 for the patient mode, there is less diversity in this mode, which may be preferable from a clinical perspective. In contrast, the cosine scores for Marble-generated factor vectors are more widely spread out, especially in the diagnosis and procedure mode. This indicates there is more overlap using Marble.

Experimentally we found Granite-generated phenotypes can cover a range of sizes for patient groups. Table III shows the phenotypes extracted using Granite, where * denotes features that were related to case patients and the † denotes features related to control patients according to our predictive model (discussed later in this section). Most phenotypes capture small parts of the population, demonstrating the potential for our algorithm to uncover rare phenotypes.

Next, we examine the clinical relevance of the generated phenotypes. A domain expert graciously annotated the Granite- and Marble-generated phenotypes as "clinically relevant", "possibly clinically relevant", and "not clinically relevant." Overall, Granite generated fewer clinically relevant phenotypes than Marble, but we found that the clinical relevance of Granite-generated phenotypes was highly correlated with the weight associated with the phenotype (i.e., higher λ_r means more likely to be relevant). On the other hand, Marble-generated phenotypes did not exhibit this relationship. Figure 5 shows the Receiver Operator Curve based on using the λ weight associated with the phenotype to classify that phenotype as clinically meaningful or not. This analysis suggests there is a trade-off between diversity and clinical relevance. By encouraging diverse solutions through the angular penalty term, Granite is more likely to find less relevant phenotypes that correspond to smaller weights, and in practice these phenotypes can be discarded. Moreover, the discriminative power of Granite and its ability to generate sparse and diverse phenotypes make it a useful tool for clinicians.

Finally, we compare Granite's predictive performance to Marble, CP-APR with nonnegative constraints, CP-ALS with nonnegative constraints, and Nonnegative Matrix Factorization (NMF) using a classification task to predict resistant hypertension patients. It is important to note that the derived features for these methods are obtained through unsupervised learning (i.e., phenotypes are not adapted to fit the classification model). For the five methods, we fix the number of computational phenotypes at thirty ($R = 30$), based on an analysis of the log-likelihood, and derive computational phenotypes from the constructed tensor. We performed a grid search on parameters for Granite and Marble in order to

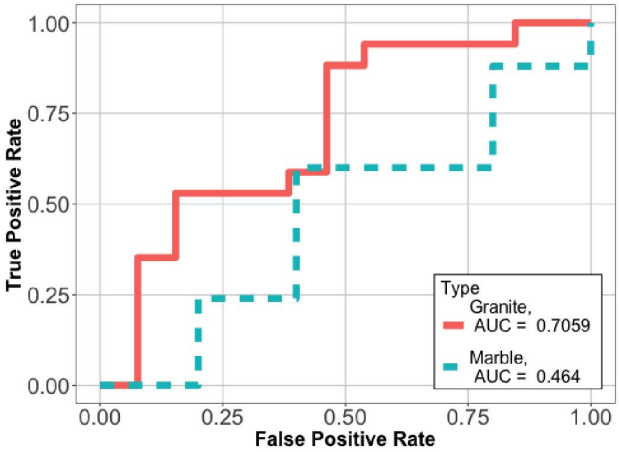
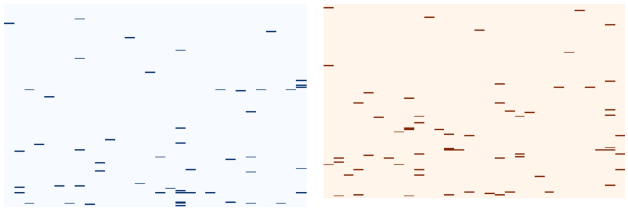
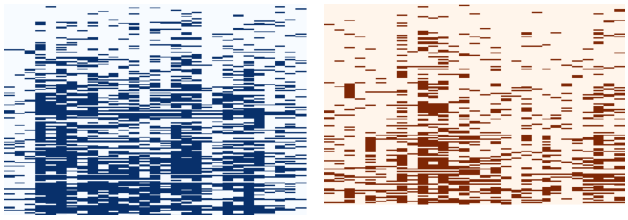


Fig. 5: ROC for Granite and Marble where task was to predict which phenotypes are clinically significant based on λ weight.



(a) Granite-generated phenotypes



(b) CP-APR-generated phenotypes

Fig. 6: Heatmap of non-zero elements in factors of diagnosis (dark blue) and medication (dark orange) modes generated by Granite and CP-APR phenotypes.

obtain a good tradeoff between sparsity and diversity. We then train ℓ_1-regularized logistic regression models on phenotypes from each of the aforementioned methods. Note for NMF, phenotypes are derived from a matricized version of the tensor (i.e., $\mathbf{W}$ are the features where $\mathbf{X} \approx \mathbf{W}\mathbf{H}^\intercal$). We ran the model on five 80-20 train-test splits, generated using stratified random sampling, with the features derived from the training dataset only. For CP-ALS, CP-APR, Marble, and Granite, the phenotype membership matrix is the feature matrix, and for NMF, the patient loadings matrix is the feature matrix. The optimal LASSO parameter for the regression model is learned via 10-fold cross-validation in the SD population.

Table II shows the area under the receiver operating characteristic curve (AUC) for the different methods and the average number of non-zero entries in the diagnosis and medication modes per phenotype. Granite has a higher predictive perfor-

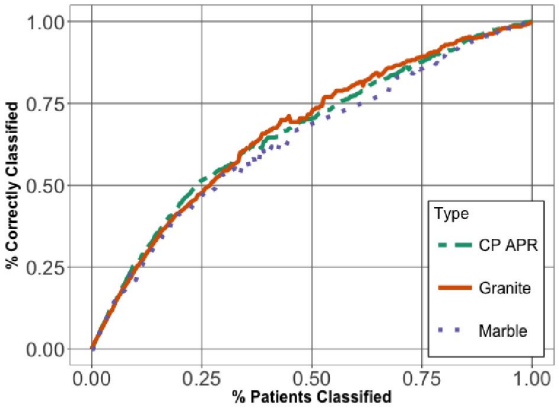


Fig. 7: Cumulative gains chart for predicting hypertension case and controls.

TABLE II: AUC for classification task (run with parameter values listed in Table I).

Method	AUC	Std. Dev.	Non-zeros / Phenotype
Granite	0.7298	0.0243	**4.6300** (w/o bias)
Marble	0.7197	0.0190	5.3330 (w/o bias)
CP-APR	0.7405	0.0117	111.0000
CP-ALS	0.6765	0.0234	113.1522
NMF	0.7203	0.0315	NA

mance than CP-ALS, Marble, and NMF. The low performance of CP-ALS might indicate that the Poisson assumption is important. Since CP-APR is not restricted by sparsity constraints, it is able to capture more of the population and unsurprisingly has the best AUC. However, CP-APR-generated phenotypes are not sparse. Figure 6 shows the number of non-zero terms in the medication and diagnosis modes for Granite-generated phenotypes (Figure 6a) and CP-APR-generated phenotypes (Figure 6b). From a qualitative perspective, the large number of medication and diagnoses codes per phenotype (111 codes on average) of CP-APR makes the generated phenotypes harder to interpret than the substantially more concise Granite-generated phenotypes (4.6300 codes on average). Therefore, we can conclude Granite phenotypes are discriminative, sparse, and diverse, which we believe makes this method more attractive than its competitors.

To look more closely at the important features in the classification task, we return to Table III where features that are most predictive of cases and controls are indicated by * and †, respectively. It is interesting to note that in addition to "hypertension" appearing in the most predictive of features of case patients (Phenotype 9), comorbidities of hypertension, like diabetes (Phenotype 23) [29] and angina pectoris (Phenotype 21) [30], also appear to be predictive. Figure 7 shows a cumulative gains chart of Granite, Marble, and CP-APR. All three methods perform similarly on smaller proportions of the population, but, as the percent of patients classified increases, Granite is more discriminative. Granite's diverse phenotypes are expected to be more useful to clinicians because it should reduce the time needed to sift through Marble's repetitive phenotypes and CP-APR's and CP-ALS's lengthy phenotypes to discover clinically interesting features of a population.

VI. Conclusion

This paper presented Granite, a diverse and sparse Poisson nonnegative model to fit EHR count data. Our algorithm provides an unsupervised methodology to achieve high-throughput phenotyping. The model generates multiple concise and interpretable computational phenotypes with minimal supervision, but also yields high diversity factors with minimal overlapping elements between the phenotypes.

The experimental results on simulated data demonstrate the conciseness, interpretability, diversity, and predictive power of Granite-derived phenotypes. Granite can also be used to rapidly characterize, predict, and manage a large number of diverse diseases, thereby promising a novel, data-driven solution that can benefit the entire population. Despite its merits, there are certain limitations to Granite. In particular, one drawback is Granite, though it was designed as such, does not incorporate any supervision. In the future, we plan to address this issue by using weak supervision provided by outside data sources to guide the factorization to more clinically relevant phenotypes.

VII. Appendix

A. Partial derivatives of the objective function

The computation of the partial derivative for the factor vectors $\mathbf{a}_r^{(n)}$ is achieved as follows. It is useful to note that each element in the approximation tensor, $z_{\vec{i}}$, can be rewritten as follows:

$$z_{\vec{i}} = \sigma u_{i_1}^{(1)} u_{i_2}^{(2)} \cdots u_{i_N}^{(N)} + \sum_{r=1}^{R} \lambda_r a_{i_1 r}^{(1)} a_{i_2 r}^{(2)} \cdots a_{i_N r}^{(N)}$$
$$= \sigma \left(\prod_{m \neq n} u_{i_m}^{(m)} \right) u_{i_n}^{(n)} + \sum_{r=1}^{R} \lambda_r \left(\prod_{m \neq n} a_{i_m r}^{(m)} \right) a_{i_n r}^{(n)}$$

For notation purposes, we can represent the matricized form of the tensor decomposition as:

$$[\![\boldsymbol{\lambda}; \mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}]\!]_{(n)} = \boldsymbol{\lambda} \mathbf{A}^{(n)} (\mathbf{A}^{(-n)})^{\intercal}$$

where

$$\mathbf{A}^{(-n)} \equiv \mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)}$$

We first compute the gradient for the angular regularization term. We denote the cosine similarity penalty between two vectors using the function $g(\mathbf{a}_r^{(n)}, \mathbf{a}_p^{(n)})$. For convenience, we drop the n, r terms and introduce $\mathbf{b} = \mathbf{a}_p^{(n)}$ for $p \neq r$ and let $g(\mathbf{a}, \mathbf{b})$ denote the cosine similarity between two vectors $\mathbf{a}$, $\mathbf{b}$, where $g(\mathbf{a}, \mathbf{b}) = (\frac{\mathbf{b}^{\intercal}\mathbf{a}}{||\mathbf{b}||_2 ||\mathbf{a}||_2} - \theta_n)$. The gradient for the angular term is the following:

$$\frac{\partial g(\mathbf{a}, \mathbf{b})}{\partial \mathbf{a}} = \frac{\mathbf{b}||\mathbf{a}||_2^2 - < \mathbf{b}, \mathbf{a} > \mathbf{a}^{\intercal}}{||\mathbf{b}||_2 ||\mathbf{a}||_2^3}$$
$$\frac{\partial (\max\{0, g(\mathbf{a}, \mathbf{b})\})^2}{\partial \mathbf{a}} = (\max\{0, g(\mathbf{a}, \mathbf{b}))\}) \frac{\partial g(\mathbf{a}, \mathbf{b})}{\partial \mathbf{a}}$$

The partial derivative of the KL divergence step with respect to $\mathbf{a}_r^{(n)}$ is straightforward:

$$\frac{\partial \sum (z_{\vec{i}} - x_{\vec{i}} \log z_{\vec{i}})}{\partial \mathbf{a}} = [1 - \mathbf{X}_{(n)} \oslash \mathbf{Z}_{(n)}] \mathbf{a}_r^{(-n)}$$

TABLE III: Granite phenotypes ranked by λ_r, * denotes the phenotypes most predictive of being a hypertension case, † denotes the phenotypes most predictive of being a control. Diagnoses are orange (capitalized), and medications are blue (uncapitalized).

Phenotype 1 (15.43% of Patients)

Legally Blind
Major Symptoms, Abnormalities
Polyneuropathy
Cerebrovascular Disease Late Effects, Unspecified
Multiple Sclerosis
anticonvulsants
bronchodilators
anxiolytics, sedatives, and hypnotics

Phenotype 2 (10.76% of Patients)

Specified Heart Arrhythmias
Major Symptoms, Abnormalities
Heart Infection/Inflammation, Except Rheumatic
diuretics
beta-adrenergic blocking agents
antihyperlipidemic agents

Phenotype 3 (5.92% of Patients)

Other Endocrine/Metabolic/Nutritional Disorders
Severe Hematological Disorders
vitamins

Phenotype 4 (3.41% of Patients)

Rheumatoid Arthritis and Inflammatory Connective Tissue Disease
antirheumatics

Phenotype 5 (7.71% of Patients)

Other Endocrine/Metabolic/Nutritional Disorders
antihyperlipidemic agents

Phenotype 6 (0.72% of Patients)*

Lymphoma and Other Cancers
antiviral agents

Phenotype 7 (0.54% of Patients)

Severe Hematological Disorders
antiemetic/antivertigo agents

Phenotype 8 (2.24% of Patients)

Major Symptoms, Abnormalities
antifungals

Phenotype 9 (3.5% of Patients)*

Cardio-Respiratory Failure and Shock
Hypertension
antiarrhythmic agents

Phenotype 10 (0.36% of Patients)

Major Symptoms, Abnormalities
Other Infectious Diseases
antituberculosis agents

Phenotype 11 (0.54% of Patients)*

Opportunistic Infections
immunosuppressive agents
antiviral agents
antidepressants

Phenotype 12 (3.86% of Patients)

Major Symptoms, Abnormalities
Disorders of the Vertebrae and Spinal Discs
prolactin inhibitors
antiarrhythmic agents

Phenotype 13 (2.15% of Patients)

Diabetes with No or Unspecified Complications
bronchodilators
laxatives
antihistamines

Phenotype 14 (1.35% of Patients)

Other Endocrine/Metabolic/Nutritional Disorders
antiviral agents

Phenotype 15 (0.45% of Patients)†

Major Head Injury
anxiolytics, sedatives, and hypnotics
antiarrhythmic agents

Phenotype 16 (0.54% of Patients)†

Colorectal, Bladder, and Other Cancers
otic preparations
adrenal cortical steroids

Phenotype 17 (8.61% of Patients)

Pelvic Inflammatory Disease and Other Specified Female Genital Disorders
Osteoporosis and Other Bone/Cartilage Disorders
bronchodilators
anticonvulsants
vitamins
laxatives
antacids

Phenotype 18 (1.08% of Patients)

Severe Hematological Disorders
antiviral agents
antiparkinson agents
analgesics
GI stimulants
anticoagulants
chelating agents
antimetabolites

Phenotype 19 (0.72% of Patients)

Lung and Other Severe Cancers
mouth and throat products

Phenotype 20 (0.45% of Patients)†

Quadriplegia
mouth and throat products

Phenotype 21 (9.42% of Patients)*

Angina Pectoris/Old Myocardial Infarction
antianginal agents
diuretics
antiplatelet agents
nutraceutical products

Phenotype 22 (16.5% of Patients)

Precerebral Arterial Occlusion and Transient Cerebral Ischemia
Coronary Atherosclerosis/Other Chronic Ischemic Heart Disease
Urinary Tract Infection
Coagulation Defects and Other Specified Hematological Disorders
Major Symptoms, Abnormalities
Hypertension
Pressure Pre-Ulcer Skin Changes or Unspecified Stage
Other Endocrine/Metabolic/Nutritional Disorders
hormones/antineoplastics
tetracyclines
immunostimulants
antihyperlipidemic agents

Phenotype 23 (0.72% of Patients)*

Diabetes with No or Unspecified Complications
nutraceutical products

Phenotype 24 (11.03% of Patients)

Uncompleted Pregnancy With Complications
Drug/Alcohol Psychosis
Rheumatoid Arthritis and Inflammatory Connective Tissue Disease
Attention Deficit Disorder
macrolide derivatives
ophthalmic preparations

Phenotype 25 (9.15% of Patients)†

Traumatic Amputation
ophthalmic preparations
local injectable anesthetics
miscellaneous uncategorized agents

Phenotype 26 (7.89% of Patients)

Hemiplegia/Hemiparesis
hormones/antineoplastics
immunostimulants
anticonvulsants

Phenotype 28 (0.09% of Patients)

Severe Hematological Disorders
uterotonic agents

Phenotype 29 (1.17% of Patients)†

Other Eye Disorders
Poisonings and Allergic Reactions
Other Infectious Diseases
Other Endocrine/Metabolic/Nutritional Disorders
medical gas

Phenotype 30 (0.9% of Patients)

Acute Myocardial Infarction
antidiarrheals

B. *Projected Gradient Descent Line Search*

Projection for the interaction factor components $\mathbf{a}_r^{(g)}$ uses the Euclidean projection onto the ℓ_1-ball of diameter s [26] and is described by the following optimization problem:

$$\min_a \frac{1}{2}||a-b||_2^2 \qquad \text{s.t. } \sum a_i = s, a_i \geq 0. \tag{11}$$

An appropriate step size, t, is selected using backtracking line search by iteratively shrinking the step size by $\hat{\beta}_{\text{line}}$ to ensure the following condition is met:

$$f(P_\Omega(\mathbf{y} - t\nabla f(\mathbf{y}))) < f(\mathbf{y}).$$

Note that Equation (11) is the projection function, $P_\Omega(\cdot)$, in Algorithm 2. Although computing the objective function can be expensive, this ensures that our algorithm converges to a local minimum based on the standard convergence analysis of the proximal gradient method.

Algorithm 2: Projected Gradient Descent Line Search

$t = t_{\text{init}}$ # Initialize the step size
$F_t(\mathbf{y}) = \frac{1}{t}(\mathbf{y} - P_\Omega(\mathbf{y} - t\nabla f(\mathbf{y}))$
while $f(\mathbf{y} - tF_t(\mathbf{y})) > f(\mathbf{y})$ **do**
  $t = \hat{\beta}_{\text{line}} t$
  $F_t(\mathbf{y}) = \frac{1}{t}(\mathbf{y} - P_\Omega(\mathbf{y} - t\nabla f(\mathbf{y})))$
end
$\mathbf{y}^+ = P_\Omega(\mathbf{y} - t\nabla f(\mathbf{y}))$

Algorithm 3: Membership Calculation

Randomly initialize $\mathbf{B}$
for $k = 1, 2, \cdots, k_{\max}$ **do**
  Calculate $\nabla f(\mathbf{B})$
  Update $\mathbf{B}^+ = P_{\mathbf{B}}(\mathbf{B} - t\nabla f(\mathbf{B}))$
  if $|f(\mathbf{B}^+) - f(\mathbf{B})| <$ convergenceTol **then**
    break
  end
end
$\hat{\mathbf{A}}^{(1)}$ = normalize rows($\mathbf{B}$)

C. Membership of Existing Factors

Algorithm 3 details the calculation of the membership vector, $\mathbf{B}$. We define $\mathbf{B} = \boldsymbol{\alpha_m}\mathbf{I}\left[\tilde{\mathbf{A}}^{(1)} \;\; \tilde{\mathbf{u}}^{(1)}\right]$ using the factor matrices with a vector $\boldsymbol{\alpha_m}$, where m is the number of dimensions in the first axis.

ACKNOWLEDGMENT

The authors would like to thank Suriya Gunasekar for her input on inducing sparsity. This work was supported by NSF grant 1418504.

REFERENCES

[1] G. Hripcsak and D. J. Albers, "Next-generation phenotyping of electronic health records," *J. Am. Med. Inform. Assoc.*, vol. 20, no. 1, pp. 117–121, Dec. 2012.

[2] A. N. Kho, M. G. Hayes, L. Rasmussen-Torvik, J. A. Pacheco, W. K. Thompson, L. L. Armstrong, J. C. Denny, P. L. Peissig, A. W. Miller, W.-Q. Wei *et al.*, "Use of diverse electronic medical record systems to identify genetic risk for type 2 diabetes within a genome-wide association study," *J. Am. Med. Inform. Assoc.*, vol. 19, no. 2, pp. 212–218, 2012.

[3] K. M. Newton, P. L. Peissig, A. N. Kho, S. J. Bielinski, R. L. Berg, V. Choudhary, M. Basford, C. G. Chute, I. J. Kullo, R. Li, J. A. Pacheco, L. V. Rasmussen, L. Spangler, and J. C. Denny, "Validation of electronic medical record-based phenotyping algorithms: results and lessons learned from the emerge network," *J. Am. Med. Inf. Assoc.*, vol. 20, no. e1, pp. e147–e154, 2013.

[4] R. L. Richesson, J. Sun, J. Pathak, A. N. Kho, and J. C. Denny, "Clinical phenotyping in selected national networks: demonstrating the need for high-throughput, portable, and computational methods," *AI in Medicine*, vol. 71, pp. 57–61, 2016.

[5] J. C. Ho, J. Ghosh, S. R. Steinhubl, W. F. Stewart, J. C. Denny, B. A. Malin, and J. Sun, "Limestone: High-throughput candidate phenotype generation via tensor factorization." *J. Biomed. Inf.*, vol. 52, pp. 199–211, Dec. 2014.

[6] J. C. Ho, J. Ghosh, and J. Sun, "Marble: High-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization," in *Proc. of ACM Knowledge Discover and Data Mining*, 2014, pp. 115–124.

[7] S. Joshi, S. Gunasekar, D. Sontag, and J. Ghosh, "Identifiable phenotyping using constrained non-negative matrix factorization," in *1st Conf. on Machine Learning and Health Care*, 2016.

[8] J. Zhou, F. Wang, J. Hu, and J. Ye, "From micro to macro: data driven phenotyping by densification of longitudinal electronic medical records," in *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. ACM, 2014, pp. 135–144.

[9] Y. Chen, J. Ghosh, C. A. Bejan, C. A. Gunter, S. Gupta, A. Kho, D. Liebovitz, J. Sun, J. Denny, and B. Malin, "Building bridges across electronic health record systems through inferred phenotypic topics," *J. Biomed. Inf.*, vol. 55, pp. 82–93, 2015.

[10] R. Pivovarov, A. J. Perotte, E. Grave, J. Angiolillo, C. H. Wiggins, and N. Elhadad, "Learning probabilistic phenotypes from heterogeneous ehr data," *J. Biomed. Inf.*, vol. 58, pp. 156–165, 2015.

[11] R. Henao, J. T. Lu, J. E. Lucas, J. Ferranti, and L. Carin, "Electronic health record analysis via deep poisson factor models," *J. Mach. Learn. Res.*, vol. 17, no. 186, pp. 1–32, 2015.

[12] Z. Che, D. Kale, W. Li, M. T. Bahadori, and Y. Liu, "Deep computational phenotyping," in *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 507–516.

[13] D. C. Kale, Z. Che, M. T. Bahadori, W. Li, Y. Liu, and R. Wetzel, "Causal phenotype discovery via deep networks," in *Proc. of Am. Med. Inf. Assoc. Annu. Symp.*, 2015, pp. 677–686.

[14] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[15] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[16] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.

[17] E. C. Chi and T. G. Kolda, "On tensors, sparsity, and nonnegative factorizations," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 4, pp. 1272–1299, 2012.

[18] I. Perros, R. Chen, R. Vuduc, and J. Sun, "Sparse hierarchical tucker factorization and its application to healthcare," in *IEEE Int. Conf. on Data Mining*, 2015, pp. 943–948.

[19] Y. Wang, R. Chen, J. Ghosh, J. C. Denny, A. Kho, Y. Chen, B. A. Malin, and J. Sun, "Rubik: Knowledge guided tensor factorization and completion for health data analytics," in *Proc. of ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2015, pp. 1265–1274.

[20] C. Hu, P. Rai, C. Chen, M. Harding, and L. Carin, "Scalable bayesian non-negative tensor factorization for massive count data," in *Mach. Learn. Knowl. Discov. Databases*, 2015, pp. 53–70.

[21] E. Acar, M. Nilsson, and M. Saunders, "A flexible modeling framework for coupled matrix and tensor factorizations," *Proc. Eur. Signal Process. Conf. EUSIPCO*, pp. 111–115, 2014.

[22] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 67, no. 2, pp. 301–320, 2005.

[23] E. Acar, D. M. Dunlavy, and T. G. Kolda, "A scalable optimization approach for fitting canonical tensor decompositions," *Journal of J. Chemom.*, vol. 25, no. 2, pp. 67–86, 2011.

[24] A. Stegeman and P. Comon, "Subtracting a best rank-1 approximation may increase tensor rank," *Linear Algebra Appl.*, vol. 433, no. 7, pp. 1276–1300, 2010.

[25] S. Hansen, T. Plantenga, and T. G. Kolda, "Newton-based optimization for kullback-leibler nonnegative tensor factorizations," *Optim. Methods Softw.*, vol. 30, no. 5, pp. 1002–1029, 2015.

[26] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the ℓ_1-ball for learning in high dimensions." *Proc. of Int. Conf. Mach. Learn.*, pp. 272–279, 2008.

[27] D. M. Roden, J. M. Pulley, M. A. Basford, G. R. Bernard, E. W. Clayton, J. R. Balser, and D. R. Masys, "Development of a large-scale de-identified DNA biobank to enable personalized medicine," *Clin. Pharmacol. Ther.*, vol. 84, no. 3, pp. 362–369, 2008.

[28] M. D. Ritchie, J. C. Denny, D. C. Crawford, A. H. Ramirez, J. B. Weiner, J. M. Pulley, M. A. Basford, K. Brown-Gentry, J. R. Balser, D. R. Masys *et al.*, "Robust replication of genotype-phenotype associations across multiple diseases in an electronic medical record," *Am. J. Hum. Genet.*, vol. 86, no. 4, pp. 560–572, 2010.

[29] A. N. Long and S. Dagogo-Jack, "Comorbidities of diabetes and hypertension: mechanisms and approach to target organ protection," *J. Clin. Hypertens. (Greenwich)*, vol. 13, no. 4, pp. 244–251, 2011.

[30] P. Richardson and L. S. Hill, "Relationship between hypertension and angina pectoris." *Br. J. Clin. Pharmacol.*, vol. 7, no. S2, pp. 249S–253S, 1979.