Matrix Factorization as Search[⋆]

Kristian Kersting[1,2], Christian Bauckhage[1], Christian Thurau[3], Mirwaes Wahabzada[1]

[1] Fraunhofer IAIS, Schloss Birlinghoven, Sankt Augustin, Germany
[2] Institute of Geodesy and Geoinformation, University of Bonn, Germany
[3] Game Analytics Aps., Copenhagen, Denmark

Abstract. Simplex Volume Maximization (SiVM) exploits distance geometry for efficiently factorizing gigantic matrices. It was proven successful in game, social media, and plant mining. Here, we review the distance geometry approach and argue that it generally suggests to factorize gigantic matrices using search-based instead of optimization techniques.

1 Interpretable Matrix Factorization

Many modern data sets are available in form of a real-valued $m \times n$ matrix $\mathbf{V}$ of rank $r \leq \min(m, n)$. The columns $\mathbf{v}_1, \ldots, \mathbf{v}_n$ of such a data matrix encode information about n objects each of which is characterized by m features. Typical examples of objects include text documents, digital images, genomes, stocks, or social groups. Examples of corresponding features are measurements such as term frequency counts, intensity gradient magnitudes, or incidence relations among the nodes of a graph. In most modern settings, the dimensions of the data matrix are large so that it is useful to determine a compressed representation that may be easier to analyze and interpret in light of domain-specific knowledge. Formally, compressing a data matrix $\mathbf{V} \in \mathbb{R}^{m \times n}$ can be cast as a *matrix factorization* (MF) task. The idea is to determine factor matrices $\mathbf{W} \in \mathbb{R}^{m \times k}$ and $\mathbf{H} \in \mathbb{R}^{k \times n}$ whose product is a low-rank approximation of $\mathbf{V}$. Formally, this amounts to a minimization problem $\min_{\mathbf{W},\, \mathbf{H}} \left\|\mathbf{V} - \mathbf{WH}\right\|^2$ where $\|\cdot\|$ denotes a suitable matrix norm, and one typically assumes $k \ll r$.

A common way of obtaining a low-rank approximation stems from truncating the singular value decomposition (SVD) where $\mathbf{V} = \mathbf{WSU}^T = \mathbf{WH}$. The SVD is popular for it can be solved analytically and has significant statistical properties. The column vectors $\mathbf{w}_i$ of $\mathbf{W}$ are orthogonal basis vectors that coincide with the directions of largest variance in the data. Although there are many successful applications of the SVD, for instance in information retrieval, it has been criticized because the $\mathbf{w}_i$ may lack interpretability with respect to the field from which the data are drawn [6]. For example, the $\mathbf{w}_i$ may point in the direction of negative orthants even though the data itself is strictly non-negative. Nevertheless, data analysts are often tempted to reify, i.e., to assign a "physical"

[⋆] The authors would like to thank the anonymous reviewers for their comments. This work was partly supported by the Fraunhofer ATTRACT fellowship STREAM.

meaning or interpretation to large singular components. In most cases, however, this is not valid. Even if reification is justified, the interpretative claim cannot arise from mathematics, but must be based on an intimate knowledge of the application domain.

The most common way of compressing a data matrix such that the resulting basis vectors are interpretable and faithful to the data at hand is to impose additional constraints on the matrices $\mathbf{W}$ and $\mathbf{H}$. An example is *non-negative* MF (NMF), which imposes the constraint that entries of $\mathbf{W}$ and $\mathbf{H}$ are non-negative. Another example of a constrained MF method is *archetypal analysis* (AA) as introduced by [3]. It considers the NMF problem where $\mathbf{W} \in \mathbb{R}^{n \times k}$ and $\mathbf{H} \in \mathbb{R}^{k \times n}$ are additionally required to be column stochastic matrices, i.e., they are to be non-negative and each of their columns is to sum to 1. AA therefore represents every column vector in $\mathbf{V}$ as a convex combination of convex combinations of a *subset of the columns* of $\mathbf{V}$. Such constrained MF problems are traditionally solved analytically since they constitute quadratic optimization problems. Although they are convex in either $\mathbf{W}$ or $\mathbf{H}$, they are however not convex in $\mathbf{WH}$ so that we suffers from many local minima. Moreover, their memory and runtime requirements scale quadratically with the number n of data and therefore cannot easily cope with modern large-scale problems. A recent attempt to circumvent these problems is the CUR decomposition [6]. It aims at minimizing $\|\mathbf{V} - \mathbf{CUR}\|^2$ where the columns of $\mathbf{C}$ are selected from the columns of $\mathbf{V}$, the rows of $\mathbf{R}$ are selected from the rows of $\mathbf{V}$, and $\mathbf{U}$ contains scaling coefficients. Similar to AA, the factorization is expressed in terms of actual data elements and hence is readily interpretable. However, in contrast to AA, the selection is not determined analytically but by means of importance sampling from the data at hand. While this reduces memory and runtime requirements, it still requires a complete view of the data. Therefore, neither of the methods discussed so far easily applies to growing dataset that nowadays become increasingly common.

2 Matrix Factorization as Search

MF by means of column subset selection allows one to cast MF as a *volume maximization problem* rather than as norm minimization [2]. It can be shown that a subset $\mathbf{W}$ of k columns of $\mathbf{V}$ yields a better factorization than any other subset of size k, if the volume of the parallelepiped spanned by the columns of $\mathbf{W}$ exceeds the volumes spanned by the other selections. Following this line, we have recently proposed a linear time approximation for maximising the volume of the simplex $\Delta\mathbf{W}$ whose vertices correspond to the selected columns [9]. Intuitively, we aim at approximating the data by means of convex combinations of selected vectors $\mathbf{W} \subset \mathbf{V}$. That is, we aim at compressing the data such that $\mathbf{v}_i \approx \sum_{j=1}^{k} \mathbf{w}_j h_{ji}$ where $\mathbf{h}_i \succeq \mathbf{0} \;\wedge\; \mathbf{1}^T\mathbf{h}_i = 1 \quad \forall i$. Then, data vectors situated on the inside of the simplex $\Delta\mathbf{W}$ can be reconstructed perfectly, i.e., $\|\mathbf{v}_i - \mathbf{W}\mathbf{h}_i\|^2 = 0$. Accordingly, the larger the volume of $\Delta\mathbf{W}$, the better the corresponding low-rank approximation of the entire data set will be. Such volume maximization approaches are more efficient than methods based on minimizing a matrix norm. Whereas the latter requires computing both matrices $\mathbf{W}$ and $\mathbf{H}$ in every iteration, volume maximization methods compute the coefficient matrix $\mathbf{H}$ only after the matrix of basis vectors $\mathbf{W}$ has been determined. Moreover, whereas evaluating $\|\mathbf{V} - \mathbf{WH}\|^2$ is of complexity $O(n)$ for n data points $\mathbf{v}_i$, evaluating $\mathrm{Vol}(\mathbf{W})$ or $\mathrm{Vol}(\varDelta\mathbf{W})$ requires $O(k^3)$ for the $k \ll n$ currently selected columns. Moreover, transferring volume maximization from parallelepipeds to simplices has the added benefit that it allows for the use of *distance geometry*. Given the lengths $d_{i,j}$ of the edges between the k vertices of a $(k-1)$-simplex $\varDelta\mathbf{W}$, its volume $\mathrm{Vol}^k_{\varDelta\mathbf{W}}$ can be computed based on this distance information only **(*)**: $\mathrm{Vol}^k_{\varDelta\mathbf{W}} = \sqrt{\frac{-1^k}{2^{k-1}\bigl((k-1)!\bigr)^2}\det\bigl(\mathbf{A}\bigr)}$ where $\det\bigl(\mathbf{A}\bigr)$ is the *Cayley-Menger* determinant [1]. And, it naturally leads to search-based MF approaches.

A simple greedy best-first search algorithm for MF that immediately follows from what has been discussed so far works as follows. Given a data matrix $\mathbf{V}$, we determine an initial selection $X_2 = \{a, b\}$ where $\mathbf{v}_a$ and $\mathbf{v}_b$ are the two columns that are maximally far apart. That is, we initialize with the largest possible 1-simplex. Then, we consider every possible extension of this simplex by another vertex and apply **(*)** to compute the corresponding volume Vol'. The extended simplex that yields the largest volume is considered for further expansion. This process continues, until k columns have been selected from $\mathbf{V}$. Lower bounding **(*)** by assuming that all selected vertices are equidistant turns this greedy best-first into the linear time MF approach called Simplex Volume Maximization (SiVM) [9]. SiVM was proven to be successful for the fast and interpretable analysis of massive game and twitter data [7], of large, sparse graphs [8] as well as — when combined with statistical learning techniques — of drought stress of plants [4, 5]. However, we can explore and exploit the link established between MF and search even further. For instance, a greedy stochastic hill climbing algorithm (sSiVM) starts with a random initial selection of k columns of $\mathbf{V}$ and iteratively improves on it. In each iteration, a new candidate column is chosen at random and tested against the current selection: for each of the currently selected columns, we verify if replacing it by the new candidate would increase the simplex volume according to **(*)**. The column whose replacement results in the largest gain is replaced. An apparent benefit of sSiVM is that it does not require batch processing or knowledge of the entire data matrix. It allows for timely data matrix compression even if the data arrive one at a time. Since it consumes only $O(k)$ memory, it represents a truly low-cost approach to MF.

In an ongoing project on social media usage, we are running a script that constantly downloads user annotated images from the Internet. We are thus in need of a method that allows for compressing this huge collection of data in an online fashion. sSiVM appears to provide a solution. To illustrate this, we considered a standard data matrix representing Internet images collected by [10]. This publicly available data has the images re-scaled to a resolution of 32×32 pixels in 3 color channels and also provides an abstract representation using 384-dimensional GIST feature vectors. Up to when writing the present paper, sSiVM processed a stream of about 1,600,000 images (randomly selected). This

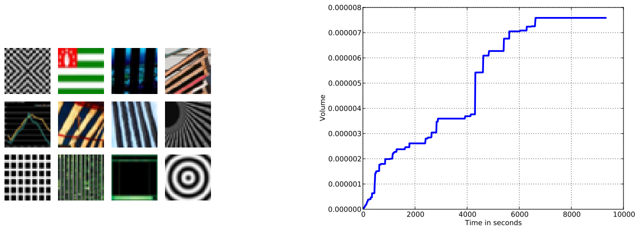

Fig. 1. (Left) Examples of 12 basis images found after 1,6 million Internet images were seen by sSiVM. (Right) Temporal evolution of the solution produced by sSiVM while computing the results shown on the left-hand side.

amounts to a matrix of 614,400,000 entries. Except for sSiVM, none of the methods discussed in this paper could reasonably handle this setting when running on a single computer. Figure 1(Left) shows a selection of 12 *basis images* obtained by sSiVM. They bear a geometric similarity to Fourier basis functions or Gabor filters. This is in fact a convincing sanity check, since GIST features are a frequency domain representation of digital images; images most similar to elementary sine or cosine functions form the extreme points in this space. Together with the measured runtime, see Fig. 1(Right), these results underline that search-based MF approaches are a viable alternative to optimization approaches.

References

1. L. M. Blumenthal. *Theory and Applications of Distance Geometry.* Oxford University Press, 1953.
2. A. Civril and M. Magdon-Ismail. On Selecting A Maximum Volume Sub-matrix of a Matrix and Related Problems. *TCS*, 410(47–49):4801–4811, 2009.
3. A. Cutler and L. Breiman. Archetypal Analysis. *Technometr.*, 36(4):338–347, 1994.
4. K. Kersting, M. Wahabzada, C. Roemer, C. Thurau, A. Ballvora, U. Rascher, J. Leon, C. Bauckhage, and L. Pluemer. Simplex distributions for embedding data matrices over time. In *SDM*, 2012.
5. K. Kersting, Z. Xu, M. Wahabzada, C. Bauckhage, C. Thurau, C. Roemer, A. Ballvora, U. Rascher, J. Leon, and L. Pluemer. Pre–symptomatic prediction of plant drought stress using dirichlet–aggregation regression on hyperspectral images. In *AAAI — Computational Sustainability and AI Track*, 2012.
6. M.W. Mahoney and P. Drineas. CUR Matrix Decompositions for Improved Data Analysis. *PNAS*, 106(3):697–702, 2009.
7. C. Thurau, K. Kersting, and C. Bauckhage. Yes We Can – Simplex Volume Maximization for Descriptive Web-Scale Matrix Factorization. In *Proc. CIKM*, 2010.
8. C. Thurau, K. Kersting, and C. Bauckhage. Deterministic CUR for improved large–scale data analysis: An empirical study. In *SDM*, 2012.
9. C. Thurau, K. Kersting, M. Wahabzada, and C. Bauckhage. Descriptive matrix factorization for sustainability: Adopting the principle of opposites. *DAMI*, 24(2):325––354, 2012.
10. A. Torralba, . Fergus, and W.T. Freeman. 80 Million Tiny Images: A Large Data Set for Nonparametric Object and Scene Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.