Rank Selection in Low-rank Matrix Approximations: A Study of Cross-Validation for NMFs

Bhargav Kanagal
Department of Computer Science
University of Maryland
College Park, MD 20770
bhargav@cs.umd.edu

Vikas Sindhwani
Mathematical Sciences
IBM T.J. Watson Research Center
Yorktown, Heights, NY 10598
vsindhw@us.ibm.com

Abstract

We consider the problem of model selection in unsupervised statistical learning techniques based on low-rank matrix approximations. While k-fold cross-validation (CV) has become the standard method of choice for model selection in supervised learning techniques, its adaptation to unsupervised matrix approximation settings has not received sufficient attention in the literature. In this paper, we emphasize the natural link between cross-validating matrix approximations and the task of matrix completion from partially observed entries. In particular, we focus on Non-negative Matrix Factorizations and propose scalable adaptations of Weighted NMF algorithms to efficiently implement cross-validation procedures for different choices of holdout patterns and sizes. Empirical observations on text modeling problems involving large, sparse document-term matrices suggest that these procedures enable easier and more accurate selection of rank (i.e., number of "topics" in text) than other alternatives for implementing CV.

1 Introduction

Let V be a large, sparse matrix of size $m \times n$ representing a collection of m high-dimensional data points in $\mathbf{R}^n$. The assumption that V is low-rank implies that most data points can be compactly and accurately represented as linear combinations of a small set of k basis vectors $H \in \mathbf{R}^{k \times n}$, with coefficients $W \in \mathbf{R}^{m \times k}$ providing a lower-dimensional encoding. When V is non-negative, it is appealing to enforce H and W to be non-negative as this lends a "parts-based" interpretability to the representation i.e., each of the non-negative data points may be seen as an additive, sparse composition of k "parts". Such Non-negative Matrix Factorizations (NMF) [5]), $V \approx WH$, find wide applicability in a variety of problems [2]. In particular, when the approximation error is measured in terms of Generalized KL-divergence (I-divergence), NMF becomes identical to probabilistic Latent Semantic Analysis (pLSI), a classic algorithm for modeling topics in text. NMF can also be regularized with different types of matrix norms, including l_1 for promoting sparsity. Similarly, with the squared frobenius approximation error, NMF is akin to Latent Semantic Analysis (LSI) except for additionally enforcing non-negativity constraints (which turns it into an NP-hard optimization problem). In this paper, we are concerned with the problem of model selection in NMFs and more specifically, how to choose the correct number of topics when building NMF models of text, using forms of cross-validation. In standard supervised learning, K-fold cross-validation refers to the common procedure where the data is partitioned into K chunks, each taking turns to serve as the heldout test set for a model trained on the rest. In the end, the error is averaged across K runs and the model with the smallest K-fold error is selected. When thinking of how to extend CV ideas to matrix approximation, several natural questions arise: How should a matrix be partitioned to define CV folds? How many folds should one use? Should rows be held out, or columns, or both? Should the held-out pattern be somehow aligned with the data sparsity to avoiding holding out too many zero cells? How should the model be trained in the presence of held out cells? The last question immedi-

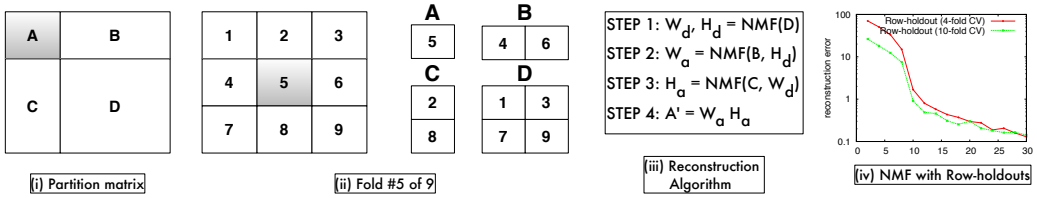

Figure 1: (i),(ii) illustrate 3×3 Gabriel holdouts. (iii) Owen and Perry's reconstruction algorithm. (iv) Row-holdouts for CV in NMF causes overfitting (monotonic decrease in reconstruction error)

ately establishes a link to matrix completion, a subject of significant interest both in the practice of recommender systems [4] as well as in the theory of nuclear norm minimization [1]. Held-out cells may be seen as missing values that need to be estimated. In this work, we propose a non-negative matrix completion based approach to implementing CV as an alternative to previously considered procedures. For scalability, we adapt weighted NMF algorithms for this purpose, where zero-valued weights allow cells to be held out. We empirically explore the distinction between *Gabriel* [3] and *Wold* [11] holdouts that correspond to holding out random cells or submatrices respectively, and study the sensitivity of rank-selection to holdout sizes.

2 Cross-Validation in Matrix Approximations

For simplicity in exposition, we work with the Frobenius distance, $\|V - WH\|^2_{fro}$, as a measure of matrix approximation error in the rest of the paper. The most common CV procedure is based on holding out rows of the matrix, as in the supervised learning case. However, models like LSI, pLSI and NMF are transductive and do not inherently allow out of sample extension, and hence a "fold-in" procedure is employed. Let V_{train} be the held-in and V_{test} be the heldout rows. An NMF is first learnt on the training set $(W_{train}, H_{train}) = \arg\min_{W,H\geq 0} \|V_{train} - WH\|^2_{fro}$. The held out error, estimated as $\min_{W\geq 0} \|V_{test} - WH_{train}\|^2$ where H_{train} is held fixed, may be seen as a measure of how well topics learnt on training rows "explain" the unseen held-out data. However, row-holds have been criticized in the literature [10] and may lead to overfitting since model parameters W are re-estimated on V_{test} and in this sense the heldout data is not completely unseen (also observed in our analysis, Figure 1(iv)).

In a recent paper, Owen and Perry [7] revisit cross-validation in the context of SVD and NMF. They use the term *Bi-Cross-Validation* (BiCV) to refer to more general two-dimensional holdout patterns. In particular, they work with *Gabriel* holdout patterns [3] where the rows of the matrix are divided into h groups and the columns are divided into l groups. The number of folds is hl; in each fold a given row and column group identifies the heldout submatrix while the remaining cells are available for training. An illustration of 3x3-fold cross-validation is shown in Figure 1(i,ii). During each round of BiCV, the matrix may be viewed as being composed of 4 submatrices, $V = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ where A is the held out block, and the training blocks B, C, D are used to reconstruct $A \approx \hat{A}$ resulting in the heldout error estimate $\|A - \hat{A}\|^2_{fro}$. BiCV for SVD and NMF calls for different reconstruction procedures. For SVD, it may be understood in terms of multivariate regression of target variables C in terms of input variables D, followed by prediction of A in terms of B. Thus, a least squares problem is solved, $\hat{\beta} = \|C - D\beta\|^2_{fro}$, which provides the estimate $\hat{A} = B\beta = BD^{\dagger}C$ where $D^{\dagger}$ is the pseudo-inverse of D. In this procedure, $D^{\dagger}$ may be *exactly* estimated from SVD of D. Thus, the BiCV procedure for SVD reduces to SVD of a training submatrix followed by reconstruction of the heldout submatrix. Owen and Perry remark that this avoids the use of missing-value methods to estimate heldout cells, for which only locally optimal solutions can be found. This is also the reason Gabriel holdouts may be preferred over *Wold* [11] holdouts where random cells (instead of submatrices) are held out and need to be estimated by missing value imputation. Owen and Perry's NMF BiCV procedure is described in Figure 1(iii). It is reasoned as follows. Instead of taking the SVD of D, an NMF $D \approx W_D H_D$ is constructed. Then, $\hat{A} = B(W_D H_D)^{\dagger} C = (BH_D^{\dagger})(W_D^{\dagger} C) = W_A H_A$ where $W_A = BH_D^{\dagger} = \arg\min_W \|B - WH_D\|^2_{fro}$ and $H_A = W_D^{\dagger} C = \arg\min_H \|C - W_D H\|^2_{fro}$. However, this reasoning produces W_A and H_A that may have negative entries and therefore Owen and Perry replace their least squares estimates with non-negative least squares (steps 2 and 3 in Figure 1(iii)).

Rather than adapting SVD BiCV formulations to NMF in this manner, we propose the direct use of non-negative matrix completion (imputation) methods. Unlike BiCV in SVD where Owen and Perry's procedure avoids potentially suboptimal imputation methods, the NMF case is different since the factorization problem itself is non-convex. Hence, we revisit imputation for both Gabriel and Wold holdouts in the NMF context. We begin by describing weighted NMFs that can be adapted for BiCV. We then show how efficient BiCV can be performed by using low-rank weight matrices. An empirical study is then reported on high dimensional text modeling problems. For related work on NMF model selection, see [8, 7] and references therein.

3 Cross-validation Using Weighted NMF

Weighted NMFs minimize the following objective function, $\arg\min_{W\geq 0, H\geq 0} \|S \odot (V - WH)\|^2$, where X is an $m \times n$ data matrix, S is an $m \times n$ matrix of "weights" and W, H are $m \times k$ and $k \times n$ matrices where $k \ll m, n$. $\odot$ is the Hadamard product. The weights S allow domain-specific emphasis in reconstructing certain matrix entries in preference to others. For example, in face recognition applications [9], more emphasis may be needed in central image pixels with higher likelihood of not being in the background. It has been shown [6, 9] that the following modifications to Lee and Seung multiplicative update rules [5] lead to minimization of the above function. ($\sqrt{S}$ is the element-wise square root, not the matrix square root):

$$W = W \odot \frac{(\sqrt{S} \odot V)H^T}{\left(\sqrt{S} \odot (WH)\right) H^T} \qquad H = H \odot \frac{W^T(\sqrt{S} \odot V)}{W^T(\sqrt{S} \odot (WH))} \tag{1}$$

In the context of cross-validation, weighted NMFs may be used by setting S to be binary, i.e., $S_{ij} = 1$ for held-in entries and 0 for held-out entries. Also note that in this case $\sqrt{S} = S$. We now develop techniques for efficiently evaluating Equation (1). Since held-in is typically a larger set than held out, we will work with $C = 1 - S$ to allow for more efficient sparse matrix computations. $S \odot V$ zeros out the held-out entries from V and can be efficiently computed using $V - C \odot V$. Computing $S \odot (WH)$ efficiently is non-trivial since it might seem that we first need to multiply W and H resulting a large dense matrix WH. In the unweighted case, $W^T(WH) = (W^TW)H$ and so we can avoid computing the dense matrix WH. However, this does not work for the weighted case. First, if C is highly sparse, we can compute

$$W^T((1 - C) \odot (WH)) = (W^TW)H - W^T(C \odot (WH))$$

where $(C \odot (WH))$ can be efficiently computed by evaluating WH only where C is sparse. However, this trick also does not work when C is large and dense such as in a 2×2 Gabriel holdout.

We now show that by setting C to be a low-rank binary matrix we can cover Gabriel holdouts in addition to a large class of Wold holdouts. First, we observe a simple relationship involving element-wise products with low-rank matrices. Let $C = \sum_{i=1}^{q} u_i v_i^T$ be a rank-q matrix where $u_i \in \Re^m$ and $v_i \in \Re^n$. Then we observe,

$$W^T(C \odot (WH)) = \sum_{i=1}^{q} W^T(D_{u_i}(WH)D_{v_i}) = \sum_{i=1}^{q} (W^T D_{u_i} W)(H D_{v_i})$$

Here, D_u is the diagonal matrix with diagonal elements given by u, i.e., $D_u(i, i) = u(i)$. Gabriel style block holdouts can be trivially covered by noting that $C = uv^T$ where $u(i) = 1$ if row i is in the heldout block, 0 otherwise and $v(i) = 1$ if column i is in the heldout block, 0 otherwise. In other words, Gabriel holdouts are a special case when $q = 1$. Taking $q > 1$, more general heldout patterns can be designed by appropriately choosing (u_i, v_i) i.e. holdout patterns consisting of multiple blocks can also be expressed.

4 Experimental Evaluation

In this section, we describe results of our preliminary experimental evaluation. To implement Wold hold outs, we split the matrix into 16 blocks (after shuffling rows and columns). We randomly select four blocks (one in each row. i.e., $q = 4$ in Section 3) to holdout and the remaining submatrices are held in. Since Weighted NMF is highly sensitive to the initial values of the W and H matrices [6], we execute it multiple times and take the mean of the reconstruction error. We work with one

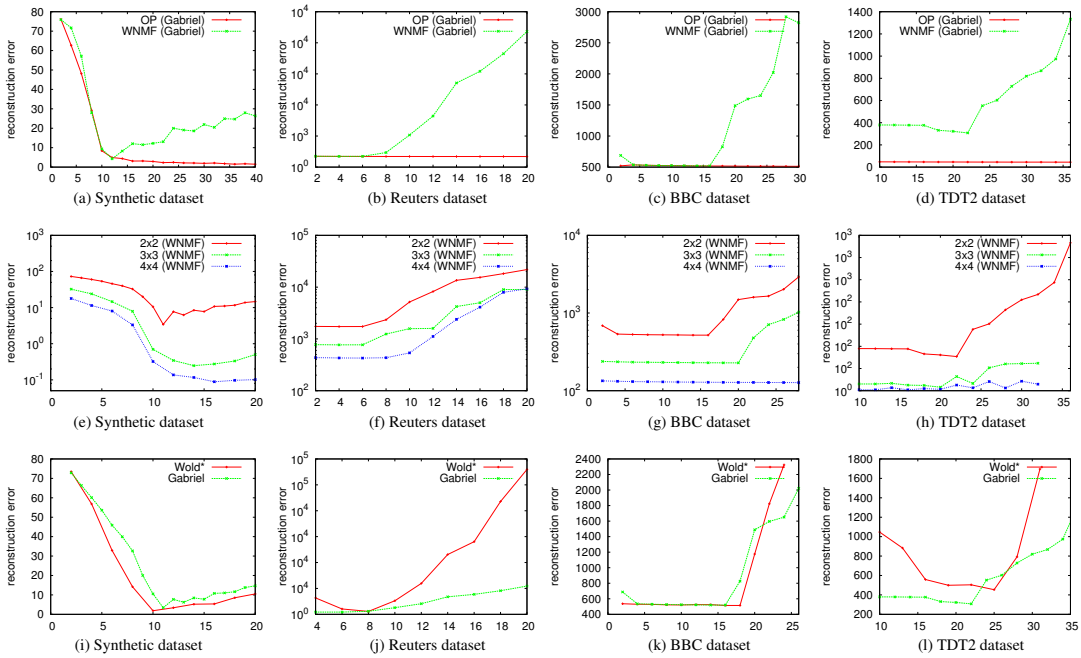

Figure 2: Results: Parts (a,b,c,d) illustrate benefits of WNMF over OP. Parts (e,f,g,h) indicate performance of different Gabriel holdouts and (i,j,k,l) compare Wold and Gabriel holdouts.

synthetic and three real world text datasets. In the **synthetic** dataset, we construct a 100×100 V matrix with rank 10, i.e., it has exactly 10 topics. We use three real world document collections with "human labeled" topics: **BBC** (size 2235×9635 with 20 topics), **Reuters** (size 7285×18221 with 10 topics), and **TDT2** (size 9394×6545 with 30 topics). We will evaluate model selection in terms of recovering the estimated number of topics in the datasets.

OP vs WNMF - for Gabriel holdouts Here, we compare the technique of Owen and Perry (OP) and our WNMF-based approach for predicting the number of topics. For both techniques, we pick the same 2×2 Gabriel holdout. We compute the cross-validation errors for different values of K for both OP and the WNMF approach; and plot them in Figure 2(a,b,c,d) for the four different datasets. As shown in the figures, for small values of K, both WNMF and OP provide almost similar reconstruction errors, however after a certain model complexity, the reconstruction error in WNMF increases dramatically. Hence, the WNMF-based technique provides a distinct point at which we can read off the model parameters; unlike OP, for which the cross-validation error decreases monotonically with increasing model complexity. For each of the datasets, we find that the minima of WNMF's reconstruction errors occurs close to the user predicted values.

Study of WNMF for different Gabriel holdouts In this experiment, we plot the cross-validation error as a function of the number of topics, for Gabriel holdouts of different sizes - 2×2, 3×3 and 4×4. The results are shown in Figure 2(e,f,g,h) for the different datasets. The errors for 4×4 holdouts is smaller than 3×3 holdouts, which in turn is smaller than 2×2 since the error is computed over a smaller set of matrix entries. As we move from 2×2 holdouts to 4×4 holdouts, the value of K at which minima is achieved progressively shifts to the right. This behavior is expected since, we are using more training data to predict a smaller heldout set – increasing model complexity can lead to a smaller reconstruction error. A similar observation was also found in Owen and Perry [7]. Also, the error curves get progressively smoother. Since we are using a larger training set, we obtain lower variance in the reconstruction error.

Gabriel vs Wold In this experiment, we check if the capability to handle more general holdouts provides a more robust procedure for model selection. We compare the cross-validation errors obtained using 2×2 Gabriel holdouts and Wold holdouts (4 iterations) for different values for K. Our results are shown in Figure 2(i,j,k,l). As shown in the Figure, the reconstruction errors for Wold holdouts are quite similar to that of Gabriel holdouts. However, the error curves are not only smoother for Wold holdouts than Gabriel holdouts, but also allow for more distinctive model selection.

References

[1] E. Candes and T. Tao. The power of convex relaxation: Near-optimal matrix completion. In *IEEE Trans. Information Theory 56(5)*, pages 2053–2080, 2009.

[2] A. Cichocki, R. Zdunek, A. Phan, and S. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley, 2009.

[3] G. K. Le biplot util dexplaration de donnees multidimensionellles. In *J. Roy. Stat. Soc. Series*, 2002.

[4] Y. Koren, R. M. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[5] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. In *Nature*, pages 788–791, 401(6755) 1999.

[6] Y. Mao and L. K. Saul. Modeling distances in large-scale networks by matrix factorization. In *Internet Measurement Conference*, pages 278–287, 2004.

[7] P. O. Perry and A. B. Owen. Bi-cross-validation of the svd and the non-negative matrix factorization. In *Annals of Applied Statistics*, pages 564–594, 2009.

[8] V. Y. F. Tan and C. Fevotte. Automatic relevance determination in nonnegative matrix factorization. In *SPARS*, 2009.

[9] N.-D. H. V. Blondel and P. V. Dooren. Weighted nonnegative matrix factorization and face feature extraction. In *submitted to Image and Vision Computing*, 2007.

[10] M. Welling, C. Chemudugunta, and N. Sutter. Deterministic latent variable models and their pitfalls. In *SDM*, 2008.

[11] H. Wold. Cross-validatory estimation of the number of components in factor and principal component models. In *Technometrics*, 1978.