Probabilistic Low-Rank Subspace Clustering

Authored by:

Shinichi Nakajima
S. D. Babacan
Minh Do

Abstract

In this paper, we consider the problem of clustering data points into low-dimensional subspaces in the presence of outliers. We pose the problem using a density estimation formulation with an associated generative model. Based on this probability model, we first develop an iterative expectation-maximization (EM) algorithm and then derive its global solution. In addition, we develop two Bayesian methods based on variational Bayesian (VB) approximation, which are capable of automatic dimensionality selection. While the first method is based on an alternating optimization scheme for all unknowns, the second method makes use of recent results in VB matrix factorization leading to fast and effective estimation. Both methods are extended to handle sparse outliers for robustness and can handle missing values. Experimental results suggest that proposed methods are very effective in clustering and identifying outliers.

1 Paper Body

Modeling data using low-dimensional representations is a fundamental approach in data analysis, motivated by the inherent redundancy in many datasets and to increase the interpretability of data via dimensionality reduction. A classical approach is principal component analysis (PCA), which implicitly models data to live in a single low-dimensional subspace within the high-dimensional ambient space. However, a more suitable model in many applications is the union of multiple low-dimensional subspaces. This modeling leads to the more challenging problem of subspace clustering, which attempts to simultaneously cluster data points into multiple subspaces and find the basis of the corresponding subspace for each cluster. Mathematically, subspace clustering can be defined as follows: Let Y be the M ? N data matrix consisting of N vectors {yi ? RM }N i=1 , which are assumed be drawn from a union of K linear (or affine) subspaces Sk of unknown dimensions dk = dim(Sk) with 0 ¡ dk ¡ M . The subspace clustering problem is to find the number of subspaces K, their dimensions {dk }K k=1 , the subspace bases, and the clustering of vectors yi into these subspaces.

Subspace clustering is widely investigated problem due to its application in a large number of fields, including computer vision [6, 12, 23], machine learning [11, 22] and system identification [31] (see [22, 28] for comprehensive reviews). Some of the common approaches include algebraicgeometric approaches such as generalized PCA (GPCA) [19, 29], spectral clustering [18], and mixture models [9, 26]. Recently, there has been a great interest in methods based on sparse and/or low-rank representation of the data [5, 7, 8, 14?17, 25]. The general approach in these methods is to first find a sparse/low-rank representation X of the data and then apply a spectral clustering method on X. It has been shown that with appropriate modeling, X provides information about the seg1

mentation of the vectors into the subspaces. Two common models for X are summarized below. ? Sparse Subspace Clustering (SSC) [7, 25]: This approach is based on representing data points yi as sparse linear combinations of other data points. A possible optimization formulation is min ?kY ? Dk2F + kD ? DXk2F + ?kXk1 , subject to diag(X) = 0 , (1) D,X

where k ? kF is the Frobenius norm and k ? k1 is the l1 -norm. ? Low-Rank Representation (LRR) [8, 14?17] : These methods are based on a principle similar to SSC, but X is modeled as low-rank instead of sparse. A general formulation for this model is min ?kY ? Dk2F + kD ? DXk2F + ?kXk? , (2) D,X

where k ? k? is the nuclear norm. In these formulations, D is a clean dictionary and data Y is assumed to be the noisy version of D possibly with outliers. When ? ? ?, Y = D, and thus the data itself is used as the dictionary [7,15,25]. If the subspaces are disjoint or independent1 , the solution X in both formulations is shown to be such that Xik 6= 0 only if data points yi and yk belong to the same subspace [7, 14, 15, 25]. That is, the sparsest/lowest rank solution is obtained when each point yi is represented as a linear combination of points in its own subspace. The estimated X is used to define an affinity matrix [18] such as —X— + —XT — and a spectral clustering algorithm, such as normalized cuts [24], is applied on this affinity to cluster the data vectors. The subspace bases can then be obtained in a straightforward manner using this clustering. These methods have also been extended to include sparse outliers. In this paper, we develop probabilistic modeling and inference procedures based on a principle similarly to LRR. Specifically, we formulate the problem using a latent variable model based on the factorized form X = AB, and develop inference procedures for estimating A, B, D (and possibly outliers), along with the associated hyperparameters. We first show a maximum-likelihood formulation of the problem, which is solved using an expectation-maximization (EM) method. We derive and analyze its global solution, and show that it is related to closed-form solution of the rankminimization formulation (2) in [8]. To incorporate automatic estimation of the latent dimensionality of subspaces and the algorithmic parameters, we further present two Bayesian approaches: The first one is based on same probability model as the EM method, but additional priors are placed on the latent variables and variational Bayesian inference is employed for approximate marginalization to avoid overfitting. The second one is based on a matrix-factorization formulation, and exploits the recent results on

Bayesian matrix factorization [20] to achieve fast estimation that is less prone to errors due to alternating optimization. Finally, we extent both methods to handle large errors (outliers) in the data, to achieve robust estimation. Compared to deterministic methods, proposed Bayesian methods have the advantages of automatically estimating the dimensionality and the algorithmic parameters. This is crucial in unsupervised clustering as the parameters can have a drastic effect on the solution, especially in the presence of heavy noise and outliers. While our methods are closely related to Bayesian PCA [2, 3, 20] and mixture models [9, 26], our formulation is based on a different model and leads to robust estimation less dependent on the initialization, which is one of the main disadvantages of such methods.

2

Probabilistic Model for Low-Rank Subspace Clustering

In the following, without loss of generality we assume that M ? N and Y is full row-rank. We also assume that each subspace is sufficiently sampled, that is, for each Si of dimension di , there exist at least di data vectors yi in Y that span Si . As for notation, the expectations are denoted by h ? i, N is the normal distribution, and diag() denotes the diagonal of a matrix. We do not differentiate the variables from the parameters of the model to have a unified presentation throughout the paper. We formulate the latent variable model as yi = di + nY , di = DAbi + nD ,

i = 1, . . . , N

(3) (4)

L PK L 1 The subspaces Sk are called independent if dim(K the direct sum. k=1 SK) = k=1 dim(Sk) with The subspaces are disjoint if they only intersect at the origin.

2

where D is M ? N , A is N ? N , and nY , nD are i.i.d. Gaussian noise independent of the data. The associated probability model is given by2

p(yi —di) = N yi — di , ?y2 IM , (5)

p(di —D, A, bi) = N di — DAbi , ?d2 IM , (6) p(bi) = N (bi —0, IN) .

(7) QN

We model the components as independent such that p(Y—D) = i=1 p(yi —di), p(D—A, B) = QN QN i=1 p(di —D, A, bi), and p(B) = i=1 p(bi). This model has the generative interpretation where latent vectors bi are drawn from an isotropic Gaussian distribution, shaped by A to obtain Abi , which then chooses a sample of points from the dictionary D to generate the ith dictionary element di . In this sense, matrix DA has a role similar to principal subspace matrix in probabilistic principal component analysis (PPCA) [26]. However, notice that in contrast to this and related approaches such as mixture of PPCAs [9, 26], the principal subspaces are defined using the data itself in (6). In (5), the observations yi are modeled as corrupted versions of dictionary elements di with iid Gaussian noise. Such separation of D and Y is not necessary if there are no outliers, as the presence of noise nY and nD makes them unidentifiable. However, we use this general formulation to later include outliers. 2.1

An Expectation-Maximization (EM) Algorithm

In (5) - (7), latent variables bi can be regarded as missing data and D, A as parameters, and an EM algorithm can be devised for their joint estimation. The complete log-likelihood is given by LC =

N X

log p(yi , bi)

(8)

i=1

with p(yi , bi) = p(yi —di) p(di —D, A, bi) p(bi). The EM algorithm can be obtained by taking the expectation of this log-likelihood with respect to (w.r.t.) B (E-step) and maximizing it w.r.t. D, A, ?d , and ?y (M-step). In the E-step, the distribution p(B—D, A, ?d2) is found as N (hBi, ?B) with hBi = ?B

1 T T A D D, ?d2

??1 B =I+

1 T T A D DA , ?d2

(9)

and the expectation of the likelihood is taken w.r.t. this distribution. In the M-step, maximizing the expected log-likelihood w.r.t. D and A in an alternating fashion yields the update equations

?1 ?1 1 1 1 T T , A = hBi hBBT i , (10) D = 2 Y 2 I + 2 h (I ? AB) (I ? AB) iB ?y ?y ?d with hBBT i = BBT + N ?B . Finally, the estimates of ?d2 and ?y2 are found as ?d2 =

kD ? DAhBik2F + N tr(AT DT DA?B) , MN

?y2 =

kY ? Dk2F . MN

(11)

In summary, the maximum likelihood solution is obtained by an alternating iterative procedure where first the statistics of B are calculated using (9), followed by the M-step updates for D, A, ?d , and ?y in (10) and (11), respectively.

2.2

Global Solution of the EM algorithm

Although the iterative EM algorithm above can be applied to estimate A, B, D, the global solutions can in fact be found in closed form. Specifically, the optimal solution is found (see the supplementary) as either AhBi = 0 or

? ?2 VT , AhBi = Vq Iq ? N ?d2 ? q q

(12)

2 Here we assume that Abi 6= wi where wi is a zero vector with 1 as the ith coefficient, to have a proper density. This is a reasonable assumption if each subspace is sufficiently sampled and the dictionary element di belongs to one of them (i.e., it is not an outlier). Outliers are explicitly handled later.

3

? ? j = max(?j , N ?d). Here, D = U?VT ? q is a q ? q diagonal matrix with coefficients ? where ? is the singular value decomposition (SVD) of D, and Vq contains ? its q right singular vectors that correspond to singular values that are larger than or equal to N ?d . Hence, the solution (12) is related to the rank-q shape interaction matrix (SIM) Vq VqT [6], while in addition it involves scaling of the singular vectors via thresholded singular values of D. Using AhBi in (10), the singular vectors of the optimal D and Y are found to be the same, and the singular values ?j of D are related to the singular values ?j of Y as ? (?j + N ?y2 ??1 N ?d j , if ?j ¿ ? ?j = (13) ?y2 +?d2 ?j ? 2 , if ?j ? N ?d d

This is a combination of two operations: down-scaling and the solutions a quadratic equation, where the latter is a polynomial thresholding operation on the singular values ?j of Y (see supplementary). Hence, the optimal D is obtained by applying the thresholding operation (13) on the singular values of Y, where the shrinkage amount is small for large singular values so that they are preserved, whereas small singular values are shrank by down-scaling. This is an interesting result, as there is no explicit penalty on the rank of D in our modeling. As shown in [8], the nuclear norm formulation (2) leads to a similar closed-form solution, but it requires the solution of a quartic equation. Finally, at the stationary points, the noise variance ?d2 is found as ?d2 =

1 N ?q

N X

?2q0 ,

(14)

q 0 =q+1

that is, the average of the squared discarded singular values of D when computing DAhBi. A simple closed form expression of ?y2 cannot be found due to the polynomial thresholding in (13), but it can simply be calculated using (11). In summary, if ?y2 and ?d2 are given, the optimal D and AhBi are found by taking the SVD of Y and applying shrinkage/thresholding operations on the singular values of Y. However, this method requires setting ?y2 and ?d2 manually. When Y itself is used as the dictionary D (i.e., ?y2 = 0), an alternative method is to choose q, the total number of independent dimensions to be retained in DAhBi, calculate ?d2 from (14), and finally use (12) to obtain AhBi. However, when ?y2 6= 0, q cannot directly be set and a trial-and-error procedure is required to find it. Although ?d2 and ?y2 can also be estimated automatically using the iterative EM procedure in Sec. 2.1, this method is susceptible to local minima, as the trivial solution AhBi = 0 also maximizes the likelihood. These issues can be overcome by employing a Bayesian estimation to automatically determine the effective dimensionality of D and AB. We develop two methods towards this goal, which are described next.

3

Variational Bayesian Low-Rank Subspace Clustering

Bayesian estimation of D, A and B can be achieved by treating them as latent variables to be marginalized over to avoid overfitting and trivial solutions such as AB = 0. Here we develop such a method based on the probability model in the previous section but with additional priors introduced on A, B and the noise variances. Before presenting our complete probability model, we first introduce the matrix-variate normal distribution as its use significantly simplifies the algorithm derivation. For a M ? N matrix X, the matrix-variate normal distribution is given by [10]

NM N M 1 T N (X—M, ?, ?) = (2?) 2 —?—? 2 —?—? 2 exp ? tr ??1 (X ?

M) ??1 (X ? M) (15) 2 where M is the mean, and ?, ? are M ? M row and N ? N column covariances, respectively. To automatically determine the number of principal components in AB, we employ an automatic relevance determination mechanism [21] on the columns of A and rows of B using priors p(A) = N (A—0, I, CA), p(B) = N (B—0, CB , I), where CA and CB are diagonal matrices with CA = diag(cA,i) and CB = diag(cB,i), i = 1, . . . , N . Jeffrey?s priors are placed on cA,i and cB,i , and they are assumed to be independent. To avoid scale ambiguity, the columns of A and rows of B can also be coupled using the same set of hyperparameters CA = CB , as in [1]. 4

For inference, we employ the variational Bayesian (VB) method [4] which leads to a fast algorithm. Let q(D, A, B, CA , CB , ?d2 , ?y2) be the distribution that approximates the posterior. The variational free energy is given by the following functional F = h log q(D, A, B, CA , CB , ?d2 , ?y2) ? log p(Y, D, A, B, CA , CB , ?d2 , ?y2)i .

(16)

Using the mean field approximation, the approximate posterior is factorized as q(D, A, B, CA , CB , ?d2 , ?y2) = q(D) q(A) q(B) q(CA) q(CB) q(?d2) q(?y2). Using the priors defined above with the conditional distributions in (5) and (6), the approximating distributions of D, A and B minimizing the free energy F are found as matrix-variate normal distributions3 q(D) = N (hDi, I, ?D), q(A) = N (hAi, ?A , ?A) and q(B) = N (hBi, ?B , I), with parameters

1 1 1 T hDi = 2 Y ?D , ??1 = IN + 2 h (I ? AB) (I ? AB) i (17) D 2 h?y i h?y i h?d i 1 1 tr(?A hBBT i) hDT Di tr(C?1 (18) ??1 A ?A) I + A = N N ?d2 1 1 ??1 tr(?A hDT Di) hBBT i tr(?A)C?1 (19) A = A + N N ?d2 1 1 T T T T (20) hAiC?1 A + 2 hD DihAihBB i = 2 hD DihBi ?d ?d 1 1 ?1 hBi = ?B 2 hAT DT Di, ??1 hAT DT DAi . (21) B = CB + h?d i h?d2 i The estimate hAi in (20) is solved using fixed-point iterations. The hyperparameter updates are given by hc?1 A,i i =

N , hAT Aiii

hc?1 B,i i =

N , diag(hBBT iii)

(22)

hkD ? DABk2F i hkY ? Dk2F i , h?y2 i = . (23) MN MN Explicit forms of the required moments are given in the supplementary. In summary, the algorithm alternates between calculating the sufficient statistics of the distributions of D, A and B, and the updates of the hyperparameters cA,i , cB,i , ?d2 and ?y2 . The convergence can be monitored during iterations using the variational free energy F. F is also useful in model comparison, which we use for detecting outliers, as explained in Sec. 5. h?d2 i =

Similarly to the matrix factorization approaches [2, 3, 13], automatic dimensionality selection is invoked via hyperparameters cA,i and cB,i , which enforce sparsity in the columns and rows of A and B, respectively. Specifically, when a particular set of variances cA,i , cB,i assume very small values, the posteriors of the ith column of A and ith row of B will be concentrated around zero, such that the effective number of principal directions in AB will be reduced. In practice, this is performed via thresholding of variances cA,i , cB,i with a small threshold (e.g., 10?10).

4

A Factorization-Based Variational Bayesian Approach

Another Bayesian method can be developed by further investigating the probability model. Essentially, the estimates of A and B is based on the factorization of D and are independent of Y. Thus, one can apply a matrix factorization method to D, and relate this factorization to DAB to find AB. Based on this idea, we modify the probabilistic model to p(D) = N (D—DL DR , I, ?12 I), d p(DL) = N (DL —0, I, CL), p(DR) = N (DR —0, CR , I), where diagonal covariances CL and CR are used to induce sparsity in the columns of DL and rows of DR , respectively. It has been shown in [20] that when variational Bayesian inference is applied to this model, the global solution is found analytically and given by DL DR = U?F VT ,

(24)

3 The optimal distribution q(A) does not have a matrix-variate normal form. However, we force it to have this form for computational efficiency (see supplementary for details).

5

where U, V contain the singular vectors of D, and ?F is a diagonal matrix, obtained by applying a specific shrinkage method to the singular values of D [20]. The number of retained singular values are therefore automatically determined. Then, setting DL DR equal to DAB, we obtain the solution T AB = Vf ??1 f ?F Vf , where the subscript f denotes the retained singular value and vectors. The only modification to the method in the previous section is to replace the estimation of A and T B in (18)-(21) with the global solution Vf ??1 f ?F Vf . Thus, this method allows us to avoid the alternating optimization for finding A and B, which potentially can get stuck in undesired local minima. Although the probability model is slightly different than the one described in the previous section, we anticipate that its global solution to be related to the factorization-based solution.

5

Robustness to Outliers

Depending on the application, the outliers might be in various forms. For instance in motion tracking applications, an entire data point might become an outlier if the tracker fails at that instance. In other applications, only a subset of coordinates might be corrupted with large errors. Both types (and possibly others) can be handled in our modeling. The only required change in the model is in the conditional distribution of the observations as p(Y—D) = N (Y—D + E, ?y2) ,

(25)

where E is the sparse outlier matrix for which we introduce the prior R C R p(E) = N (E—0, CC E , CE) = N (vec(E)—0, CE ? CE) .

(26)

R The shape of the column covariance matrix CC E and row covariance matrix CE depends on the nature C of outliers. If only entire data points might be corrupted, we can use CE = I and independent terms R R in CR E such that

CE = diag(cE,i), i = 1, . . . , N . When entire coordinates can be corrupted, C C row-sparsity in E can be imposed using CR E = I and CE = diag(cE,i). In the first case, the VB estimation rule becomes q(ei) = N (hei i, I, ?ei) with !?1 1 1 1 hei i = ?ei 2 (yi ? hdi i) ?ei = diag + , (27) h?y i h?y2 i hcR E,i i T

with the hyperparameter update hcR E,i i = hei i hei i+tr (?ei). The estimation rules for other outlier models can be derived in a similar manner. In the presence of outlier data points, there is an inherent unidentifiability between AB and E which can prevent the detection of outliers and hence reduce the performance of subspace clustering. Specifically, an outlier yi can be included in the sparse component as ei = yi or included in the dictionary D with its own subspace, which leads to (AB)ii ? 1. To avoid the latter case, we introduce a heuristic inspired by the birth and death method in [9]. During iterations, data points yi with (AB)ii larger than a threshold (e.g., 0.95) are assigned to the sparse component ei . As this might initially increase the variational energy F, we monitor its progress over a few iterations and reject this ?birth? of the sparse component if F does not decrease below its original state. This method is observed to be very effective in identifying outliers and alleviating the effect of the initialization. Finally, missing valuesQ in Y can also be handled by modifying the distribution of the observations in (5) to p(yi —di) = k?Zi N yik — dik , ?y2 , where Zi is the set containing the indices of the observed entries in vector yi . The inference procedures can be modified with relative ease to accommodate this change.

6

Experiments

In this section, we evaluate the performance of the three algorithms introduced above, namely, the EM method in Sec. 2.2, the variational Bayesian method in Sec. 3 (VBLR) and the factorizationbased method in Sec. 4 (VBLR-Fac). We also include comparisons with deterministic subspace clustering and mixture of PPCA (MPPCA) methods. In all experiments, the estimated AB matrix is used to find the affinity matrix and the normalized cuts algorithm [24] is applied to find the clustering and hence the subspaces. 6

110

LRR (? = 0.01) LRR (? = 0.16) VBLR VBLR?Fac MPPCA

Clustering accuracy (%)

100 90 80 70 60 50 40

(a)

(b)

30 0

Figure 1: Clustering 1D subspaces (points in the same cluster are in the same color) (a) MPPCA [3] result, (b) the result of the EM algorithm (global solution). The Bayesian methods give results almost identical to (b).

20

40 60 80 Percentage of Outliers (%)

100

Figure 2: Accuracy of clustering 5 independent subspaces of dimension 5 for different percentage of outliers.

Synthetic Data. We generated 27 line segments intersecting at the origin, as shown in Fig. 1, where each contains 800 points slightly corrupted by iid Gaussian noise of variance 0.1. Each line can be considered as a separate 1D subspace, and the subspaces are disjoint but not independent. We first applied the mixture of PPCA [3] to which we provided the dimensions and the number of the subspaces. This method is sensitive to the proximity of the subspaces, and in all of our trials gave results similar to Fig. 1(a), where close lines are clustered together. On the other hand, the EM method accurately clusters the lines into different subspaces (Fig. 1(b)), and it is extremely efficient involving only one SVD. Both Bayesian methods VBLR and VBLR-Fac gave similar results and accurately estimated the subspace dimensions, while the VB-variant of MPPCA [9] gave results similar to Fig. 1(a). Next, similarly to the setup in [15], we construct 5 independent subspaces {Si } ? R50 of dimension 5 with bases Ui generated as follows: We first generate a random 50 ? 5 orthogonal matrix U1 , and then rotate it with random orthonormal matrices Ri to obtain Ui = Ri U1 , i = 2, 3, 4. Dictionary D is obtained by sampling 25 points from each subspace using Di = Ui Vi where Vi are 5 ? 25 matrices with elements drawn from N (0, 1). Finally, Y is obtained by corrupting D with outliers sampled from N (0, 1) and normalized to lie on the unit sphere. We applied our methods VBLR and VBLR-Fac to cluster the data into 5 groups, and compare their performance with MPPCA and LRR. Average clustering errors (over 20 trials) in Fig. 2 show that LRR and the proposed methods provide much better performance than MPPCA. VBLR and VBLR-Fac gave similar results, while VBLR-Fac converges much faster (generally about 10 vs 100 iterations). Although LRR also gives very good results, its performance varies with its parameters. As an example, we included its results obtained by the optimal and a slightly different parameter value, where in the latter case the degradation in accuracy is evident.

Table 1: Clustering errors (%) on the Hopkins155 motion database

Method	Mean	Max	Std
GPCA [19]	30.51	55.67	11.79
LSA [30]	8.77	38.37	9.80
SSC [7]	3.66	37.44	7.21
LRR [15]	1.71	32.50	4.85
VBLR	1.75	35.13	4.92
VBLR-Fac	1.85	37.32	5.10

Real Data with Small Corruptions. The Hopkins155 motion database [27] is frequently used to test subspace clustering methods. It consists of 156 sequences where each contains 39 to 550 data vectors corresponding to either 2 or 3 motions. Each motion corresponds to a subspace and each sequence is regarded as a separate clustering task. While most existing methods use a pre-processing stage that generally involves dimensionality reduction using PCA, we do not employ pre-processing and apply our Bayesian methods directly (the EM method cannot handle outliers and thus is not included in the experiments). The mean and maximum clustering errors and the standard deviation in the whole set are shown in Table 1. The proposed methods provide close to state-of-the-art performance, while competing methods require manual tuning of their parameters, which can affect their performance. For instance, the results of LRR is obtained by setting its parameter ? = 4, while changing it to ? = 2.4 gives 3.13% error [15]. The method in [8], which is similar to our EM7

method except that it also handles outliers, achieves an error rate of 1.44%. Finally, the deterministic method [17] achieves an error rate of 0.85% and to our knowledge, is the best performing method in this dataset. Real Data with Large Corruptions. To test our methods in real data with large corruptions, we use the Extended Yale Database B [12] where we chose the first 10 classes that contain 640 frontal face images. Each class contains 64 images and each image is resized to 48 ? 42 and stacked to generate the data vectors. Figure 3 depicts some example images, where significant corruption due to shadows and heavy noise is evident. The task is to cluster the 640 images into 10 classes. The segmentation accuracies achieved by the proposed methods and some existing methods are listed in Table 2, where it is evident that the proposed methods achieve state-of-art-performance. Example recovered clean dictionary and sparse outlier components are shown in Fig. 3. Table 2: Clustering accuracy (%) on the Extended Yale Database B Method Average

Y

LSA [30] 31.72

SSC [7] 37.66

VBLR DAB

LRR [15] 62.53

E

VBLR 69.72

VBLR-Fac 67.62

VBLR-Fac DAB E

Figure 3: Examples of recovered clean data and large corruptions. Original images are shown in the left column (denoted by Y), the clean dictionary elements obtained by VBLR and VBLR-Fac are shown in columns denoted by DAB, and columns denoted by E show corruption captured by the sparse element.

7

Conclusion

In this work we presented a probabilistic treatment of low dimensional subspace clustering. Using a latent variable formulation, we developed an expectation-maximization method and derived its global solution. We further proposed two effective Bayesian methods both based on the automatic relevance determination principle and variational Bayesian approximation for inference. While the first one, VBLR, relies completely on alternating optimization, the second one, VBLR-Fac, makes use of the global solution of VB matrix factorization to eliminate one alternating step and leads to faster convergence. Both methods have been extended to handle sparse large corruptions in the data for robustness. These methods are advantageous over deterministic methods as they are able to automatically determine the total number of principal dimensions and all required algorithmic parameters. This property is particularly important in unsupervised settings. Finally, our formulation can potentially be extended for modeling multiple nonlinear manifolds, by the use of kernel methods. Acknowledgments. The authors thank anonymous reviewers for helpful comments. SDB acknowledges the Beckman Institute Postdoctoral Fellowship. SN thanks the support from MEXT Kakenhi 23120004. MND was partially supported by NSF CHE 09-57849. 8

2 References

[1] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos. Sparse Bayesian methods for low-rank matrix estimation. IEEE Trans. Signal Proc., 60(8), Aug 2012. [2] C. M. Bishop. Bayesian principal components. In NIPS, volume 11, pages 382?388, 1999. [3] C. M. Bishop. Variational principal components. In Proc. of ICANN, volume 1, pages 514?509, 1999. [4] C.M. Bishop. Pattern Recognition and Machine Learning. Springer, 2006. [5] E. J. Cand'es, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? CoRR, abs/0912.3599, 2009. [6] J. P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. Int. J. Comput. Vision, 29(3):159?179, September 1998. [7] E. Elhamifar and R. Vidal. Sparse subspace clustering. In CVPR, pages 2790?2797, 2009. [8] P. Favaro, R. Vidal, and A. Ravichandran. A closed form solution to robust subspace estimation and clustering. In CVPR, pages 1801?1807, 2011. [9] Z. Ghahramani and M. J. Beal. Variational inference for Bayesian mixtures of factor analysers. In NIPS, volume 12, pages 449?455, 2000. [10] A. K. Gupta and D. K. Nagar. Matrix Variate Distributions. Chapman & Hall/CRC, New York, 2000. [11] K. Huang and S. Aviyente. Sparse representation for signal classification. In NIPS, 2006. [12] K.-C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. IEEE Trans. Pattern Anal. Machine Intell., 27:684?698, 2005. [13] Y. J. Lim and T. W. Teh. Variational Bayesian approach to movie rating prediction. In Proc. of KDD Cup and Workshop, 2007. [14] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. CoRR, abs/1010.2955, 2012. [15] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In ICML, pages 663?670, 2010. [16] G. Liu, H. Xu, and S. Yan. Exact subspace segmentation and outlier detection by low-rank representation. In AISTATS, 2012. [17] G. Liu and S. Yan. Latent low-rank representation for subspace segmentation and feature extraction. In ICCV, 2011. [18] U. Luxburg. A tutorial on spectral clustering. Statistics and Computing, 17(4):395?416, December 2007. [19] Y. Ma, A. Yang, H. Derksen, and R. Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data,. SIAM Review, 50(3):413?458, 2008. [20] S. Nakajima and M. Sugiyama. Theoretical analysis of Bayesian matrix factorization. Journal of Machine Learning Research, 12:2583?2648, 2011. [21] R. M. Neal. Bayesian Learning for Neural Networks. Springer, 1996. [22] H. Peterkriegel, P. Kroger, and A. Zimek. Clustering high-dimensional data: a survey on subspace slustering, pattern-based clustering, and correlation clustering. In Proc. KDD,

2008. [23] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. IEEE Trans. Pattern Anal. Machine Intell., 32(10):1832?1845, 2010. [24] J. Shi and J. Malik. Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Machine Intell., 22(8):888 ?905, aug 2000. [25] M. Soltanolkotabi and E. J. Cand'es. A geometric analysis of subspace clustering with outliers. CoRR, 2011. [26] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analyzers. Neural Comput., 11(2):443?482, February 1999. [27] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In CVPR, June 2007. [28] R. Vidal. Subspace clustering. IEEE Signal Process. Mag., 28(2):52?68, 2011. [29] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). IEEE Trans. on PAMI, 27(12):1945?1959, 2005. [30] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In ECCV, volume 4, pages 94?106, 2006. [31] C. Zhang and R. R. Bitmead. Subspace system identification for training-based MIMO channel estimation. Automatica, 41:1623?1632, 2005.

9