
Data centering and low-rank approximation

Abstract

Low-rank approximation, better known in the machine learning literature as principal component analysis, is a data modeling tool. Data centering, on the other hand, is a common preprocessing step. This paper studies the combination of low-rank approximation with data centering. In the case of no weights and no constraints, the common preprocessing practice of mean subtraction leads to optimal results and therefore need not be changed. The formal proof of this fact, however, reveals that preprocessing by mean subtraction is not the only way to optimally preprocess the data. In the case of weighted and/or constrained low-rank approximation problems, the common preprocessing practice leads to suboptimal results and therefore should be changed. The paper shows, how classical solution methods for weighted and structured low-rank approximation can be modified for doing optimal preprocessing at the same time as low-rank approximation.

Keywords: low-rank approximation, errors-in-variables modeling, data preprocessing, matrix centering, affine model.

1 Introduction

Affine data modeling

Consider a set of observed variables d_1, \dots, d_q and let $d := \text{col}(d_1, \dots, d_q)$ be the column vector of these variables. We say that the variables d_1, \dots, d_q satisfy a *linear static model* if $d \in \mathcal{L}$, where \mathcal{L} , the model, is a subspace of \mathbb{R}^q . The *complexity* of a linear model is measured by its dimension. Of interest is data fitting by low complexity models, in which case, generally, the model may only fit approximately the data. Approximate linear models for a given set of

data points $\mathcal{D} = \{d^{(1)}, \dots, d^{(N)}\} \subset \mathbb{R}^q$ are computed by low-rank approximation of the data matrix $D := [d^{(1)} \ \dots \ d^{(N)}]$. The classical approach for data fitting involves, in addition, a priori chosen input/output partition of the variables $\text{col}(a, b) := \Pi d$, where Π is a permutation matrix. Then the low-rank approximation problem reduces to the problem of solving approximately an overdetermined system of equations $AX \approx B$ (i.e., regression), where $[A \ B] := (\Pi D)^\top$. By choosing specific fitting criteria, the classical approach leads to well-know optimization problems, e.g., linear least squares, total least squares, robust least squares, and their numerous variations.

Closely related to the linear model is the affine one. We say that the variables d_1, \dots, d_q satisfy an *affine static model* \mathcal{A} if $d \in \mathcal{A}$, where \mathcal{A} is an affine set, i.e., $\mathcal{A} = c + \mathcal{L}$, with \mathcal{L} a subspace of \mathbb{R}^q and $c \in \mathbb{R}^q$ an offset vector. Obviously the affine model class contains as a special case the linear model class. The parameter c , however, allows us to account for a constant offset in the data. Consider, for example, the data $d^{(1)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $d^{(2)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, which satisfies the affine model $\mathcal{A} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \{d \mid [1 \ 0] d = 0\}$ but is not fitted by a linear model of dimension one.

The two-stage procedure

Subtracting off the offset c from the data vector d , reduces the affine modeling problem with known offset parameter to an equivalent linear modeling problem. In a realistic data modeling setup, however, the offset parameter is unknown and has to be identified together with the subspace \mathcal{L} . An often used heuristic for solving this problem is to replace the offset c by the mean $(d^{(1)} + \dots + d^{(N)})/N$ of the data. This leads to the following two-stage procedure for identification of affine models:

1. *pre-processing step*: subtract the mean from the data points,
2. *linear identification step*: identify a linear model for the centered data.

When the aim is to derive an optimal in some specified sense approximate affine model, the two-stage procedure may lead to suboptimal results. Indeed, even if the data centering and linear identification steps are

individually optimal with respect to the desired optimality criterion, their composition need not be optimal for the affine modeling problem, *i.e.*, simultaneous subspace fitting and centering.

Example 1 (Affine modeling by output error minimization). In order to illustrate the suboptimality of the two-stage procedure in a specific well-known data modeling approach, consider the overdetermined system of equations $Ax \approx b$ (linear regression). It can be shown that the variables corresponding to A are the model inputs and the variable corresponding to b is the model output. Affine model identification by output error minimization, also known as data fitting with an intercept, is the following optimization problem

$$\begin{aligned} & \text{minimize} && \text{over } \mu, x \text{ and } \Delta b && \|\Delta b\| \\ & \text{subject to} && Ax = b - \mu \mathbf{1}_N + \Delta b. \end{aligned} \quad (1)$$

Here $\mathbf{1}_N$ is the vector in \mathbb{R}^N with all elements equal to one. (The offset parameter c from the discussion above is $c = \text{col}(0, \mu)$ in (1).) A trivial modification makes (1) a standard linear least-squares problem, which solution (assuming $[A \ \mathbf{1}_N]$ is full column rank) is

$$\begin{bmatrix} x^* \\ \mu^* \end{bmatrix} = \left([A \ \mathbf{1}_N]^\top [A \ \mathbf{1}_N] \right)^{-1} [A \ \mathbf{1}_N]^\top b.$$

The alternative two-stage procedure computes, on the first stage, the mean

$$\mu_{\text{two-stage}} := (b_1 + \dots + b_N)/N.$$

Note that the mean $\mu_{\text{two-stage}}$ is the solution of the optimization problem

$$\begin{aligned} & \text{minimize} && \text{over } \mu \text{ and } \Delta b && \|\Delta b\| \\ & \text{subject to} && b - \mu \mathbf{1}_N = \Delta b, \end{aligned} \quad (2)$$

i.e., the least squares problem (1) with x set to zero. On the second stage, the value of the x parameter is determined from the least squares problem

$$Ax = b - \mu_{\text{two-stage}} \mathbf{1}_N + \Delta b$$

i.e., from (1) with μ set to $\mu_{\text{two-stage}}$. Then

$$x_{\text{two-stage}} := (A^\top A)^{-1} A^\top (b - \mu_{\text{two-stage}} \mathbf{1}_N),$$

In general, $\mu_{\text{two-stage}}$ and $x_{\text{two-stage}}$ are not equal to μ^* and x^* , respectively. The result of a numerical example with 5 data points in \mathbb{R}^2 is plotted in Figure 1 and shows that the heuristic behind the two-stage procedure may be “far” from optimal.

In the output error case, the direct approach is as difficult as the heuristic approach (linear least squares), so that there is no reason to use the heuristic on the first place. Still preprocessing by subtraction of the mean is a common practice in data modeling even in cases when the procedure is *known* to be suboptimal.

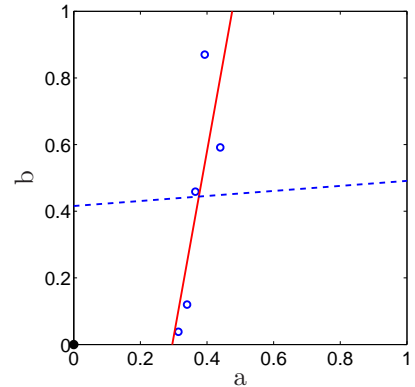


Figure 1: Example of data fitting with an intercept: **solid** — direct solution, **dashed** — two-stage solution.

Contribution of the paper

It is not clear a priori whether the two-stage procedure is optimal when combined with other data modeling approaches, such as low-rank approximation.

- We prove that in the case of low-rank approximation in the Frobenius norm with no additional constraints (*i.e.*, PCA (Jolliffe, 2002; Bishop, 2006)), the two-stage procedure is optimal. It follows from the analysis that a solution is not unique. The fact that the two-stage procedure is optimal when combined with PCA is mentioned in (Wentzell *et al.*, 1997, Section 3.4) and (Zhang and Zha, 2005, pages 315 and 316)), however, we are not aware of a formal proof in the literature. The characterization of the nonuniqueness of the solution seems to be new. The same source of nonuniqueness holds as well in the case of weighted low-rank approximation, although the two-stage procedure is suboptimal in that case.
- We show by counter examples that in the more general cases of weighted and Hankel structured low-rank approximation problems (*i.e.*, modified PCA problems), the two-stage procedure is suboptimal and propose iterative methods for solving the problem in these cases. The methods are based on the well-known alternating projections and variable projections algorithms and inherit the nice properties of these algorithms (global convergence to a locally optimal solution).

Weighted and structured low-rank approximation problems have applications in chemometrics, signal processing, and system identification (Markovsky *et al.*, 2006). The extension to weighted norms is needed in case the data is perturbed by noise with a known covariance matrix that is not a multiple of the iden-

tity. The appropriate statistical model is called errors-in-variables (Gleser, 1981). The extension to Hankel structured data matrices is motivated by their close relation to linear-time invariant dynamical models, see Section 5.

2 Preliminaries and notation

Matrix centering

Given a matrix $D \in \mathbb{R}^{q \times N}$, we define the (column¹) mean of D

$$\mathcal{M}(D) := \frac{1}{N} D \mathbf{1}_N = (d^{(1)} + \dots + d^{(N)})/N \in \mathbb{R}^q.$$

The corresponding matrix centering operation is to subtract the mean from all columns of D :

$$\mathcal{C}(D) := D - \mathcal{M}(D) \mathbf{1}_N^\top = D \left(I - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^\top \right).$$

The following proposition justifies the name ‘‘matrix centering’’ for $\mathcal{C}(\cdot)$.

Proposition 2. *The matrix $\mathcal{C}(D)$ is column centered, i.e.,*

$$\mathcal{M}(\mathcal{C}(D)) = 0.$$

Proof.

$$\begin{aligned} \mathcal{M}(\mathcal{C}(D)) &= \mathcal{M}\left(D - \mathcal{M}(D) \mathbf{1}_N^\top\right) \\ &= \frac{1}{N} \left(D - \frac{1}{N} D \mathbf{1}_N \mathbf{1}_N^\top \right) \mathbf{1}_N \\ &= \frac{1}{N} D \mathbf{1}_N - \frac{1}{N^2} D \mathbf{1}_N \underbrace{\mathbf{1}_N^\top \mathbf{1}_N}_N = 0. \end{aligned}$$

□

Next we give an interpretation of the mean computation as a simple optimal modeling problem.

Proposition 3 (Mean computation as an optimal modeling). *$\mathcal{M}(D)$ is solution of the following optimization problem:*

$$\begin{aligned} &\text{minimize over } \widehat{D}, c \quad \|D - \widehat{D}\|_F \\ &\text{subject to} \quad \widehat{D} = c \mathbf{1}_N^\top. \end{aligned}$$

Proof. The optimization problem is a linear least squares problem and its solution is

$$\widehat{c} = D \mathbf{1}_N (\mathbf{1}_N^\top \mathbf{1}_N)^{-1} = \frac{1}{N} D \mathbf{1}_N = \mathcal{M}(D).$$

□

¹The case of row-wise centering reduces to the case of column-wise centering by transposing the data matrix.

Note 4 (Intercept). Data fitting with an intercept is a special case of centering when all but one of the row means are set to zero (see Example 1), i.e., centering of one row. Intercept is appropriate when an input/output partition of the variables is imposed and there is a single output that has an offset.

Kernel and image representations of static linear models

A static linear model \mathcal{L} with q variables is a subset of \mathbb{R}^q . Two basic representations of \mathcal{L} , used in the paper, are

- kernel representation:

$$\mathcal{L} = \ker(R) := \{ d \in \mathbb{R}^q \mid R d = 0 \},$$

with parameter $R \in \mathbb{R}^{p \times q}$,

- image representation:

$$\mathcal{L} = \text{image}(P) := \{ d = P \ell \mid \ell \in \mathbb{R}^m \},$$

with parameter $P \in \mathbb{R}^{q \times m}$.

3 Unweighted low-rank approximation with centering

In this section, we consider the low-rank approximation problem in the Frobenius norm with centering:

$$\begin{aligned} &\text{minimize over } \widehat{D}, c \quad \|D - c \mathbf{1}_N^\top - \widehat{D}\|_F \\ &\text{subject to} \quad \text{rank}(\widehat{D}) \leq m. \end{aligned} \quad (3)$$

The following theorem shows that the two-stage procedure yields a solution to (3).

Theorem 5 (Optimality of the two-stage procedure). *A solution to (3) is the mean of D , $c^* = \mathcal{M}(D)$, and an optimal in a Frobenius norm rank- m approximation \widehat{D}^* of the centered data matrix $\mathcal{C}(D)$.*

Proof. Using a kernel representation of the rank constraint

$$\begin{aligned} \text{rank}(\widehat{D}) \leq m \\ \iff \text{there is full rank matrix } R \in \mathbb{R}^{(q-m) \times q} \\ \text{such that } R \widehat{D} = 0, \end{aligned}$$

we have the following equivalent problem to (3)

$$\begin{aligned} &\text{minimize over } \widehat{D}, c, R \quad \|D - c \mathbf{1}_N^\top - \widehat{D}\|_F^2 \\ &\text{subject to} \quad R \widehat{D} = 0 \quad \text{and} \quad R R^\top = I_{q-m}. \end{aligned} \quad (4)$$

The Lagrangian of (4) is

$$L(\widehat{D}, c, R, \Lambda, \Xi) := \sum_{i=1}^q \sum_{j=1}^N (d_{ij} - c_i - \widehat{d}_{ij})^2 + 2 \operatorname{trace}(R\widehat{D}\Lambda) + \operatorname{trace}(\Xi(I - RR^\top)).$$

Setting the partial derivatives of L to zero, we obtain the necessary optimality conditions

$$\partial L / \partial \widehat{D} = 0 \implies D - c\mathbf{1}_N^\top - \widehat{D} = R^\top \Lambda^\top, \quad (5)$$

$$\partial L / \partial c = 0 \implies Nc = (D - \widehat{D})\mathbf{1}_N, \quad (6)$$

$$\partial L / \partial R = 0 \implies \widehat{D}\Lambda = R^\top \Xi, \quad (7)$$

$$\partial L / \partial \Lambda = 0 \implies R\widehat{D} = 0, \quad (8)$$

$$\partial L / \partial \Xi = 0 \implies RR^\top = I. \quad (9)$$

The theorem follows from the system of equations (5–9). Next we list the derivation steps.

From (7), (8), and (9), it follows that $\Xi = 0$ and from (5), we obtain

$$D - \widehat{D} = c\mathbf{1}_N^\top + R^\top \Lambda^\top.$$

Substituting the last identity in (6), we have

$$Nc = (c\mathbf{1}_N^\top + R^\top \Lambda^\top)\mathbf{1}_N = Nc + R^\top \Lambda^\top \mathbf{1}_N \implies R^\top \Lambda^\top \mathbf{1}_N = 0 \implies \Lambda^\top \mathbf{1}_N = 0.$$

Multiplying (5) from the left by R and using (8) and (9), we have

$$R(D - c\mathbf{1}_N^\top) = \Lambda^\top. \quad (10)$$

Now, multiplication of the last identity from the right by $\mathbf{1}_N$ and use of $\Lambda^\top \mathbf{1}_N = 0$, shows that c is the row mean of the data matrix D ,

$$R(D\mathbf{1}_N - Nc) = 0 \implies c = \frac{1}{N}D\mathbf{1}_N.$$

Next, we show that \widehat{D} is an optimal in a Frobenius norm rank- m approximation of $D - c\mathbf{1}_N^\top$. Multiplying (5) from the right by Λ and using $\widehat{D}\Lambda = 0$, we have

$$(D - c\mathbf{1}_N^\top)\Lambda = R^\top \Lambda^\top \Lambda. \quad (11)$$

Defining $\Sigma := \sqrt{\Lambda^\top \Lambda}$ and $V := \Lambda \Sigma^{-1}$, (10) and (11) become

$$R(D - c\mathbf{1}_N^\top) = \Sigma V^\top, \quad V^\top V = I \\ (D - c\mathbf{1}_N^\top)V = R^\top \Sigma, \quad RR^\top = I.$$

The above equations show that the rows of R and the columns of V span, respectively, left and right m -dimensional singular subspaces of the centered data

matrix $D - c\mathbf{1}_N^\top$. The optimization criterion is minimization of

$$\|D - \widehat{D} - c\mathbf{1}_N^\top\|_F = \|R^\top \Lambda^\top\|_F = \sqrt{\operatorname{trace}(\Lambda \Lambda^\top)} = \operatorname{trace}(\Sigma).$$

Therefore, a minimum is achieved when the rows of R and the columns of V span the, respectively left and right m -dimensional singular subspaces of the centered data matrix $D - c\mathbf{1}_N^\top$, corresponding to the m smallest singular values. The solution is unique if and only if the m th singular value is strictly bigger than the $(m+1)$ st singular value. Therefore, \widehat{D} is a Frobenius norm optimal rank- m approximation of the centered data matrix $D - c\mathbf{1}_N^\top$, where $c = D\mathbf{1}_N/N$. \square

The result of Theorem 5 is a common knowledge, see, *e.g.*, (Wentzell *et al.*, 1997, Section 3.4) and (Zhang and Zha, 2005, pages 315 and 316)), however, we are not aware of a formal proof in the literature. Apart from filling this gap, the proof of Theorem 5, reveals nonuniqueness of the solution produced by the two-stage procedure (as well as by any other procedure).

Theorem 6 (Nonuniqueness). *Let*

$$\widehat{D} = PL, \quad \text{where } P \in \mathbb{R}^{q \times m} \text{ and } L \in \mathbb{R}^{m \times N}$$

be a rank revealing factorization of an optimal in a Frobenius norm rank- m approximation of the centered data matrix $\mathcal{C}(D)$. The solutions of (3) are of the form

$$\begin{aligned} c^*(z) &= \mathcal{M}(D) + Pz \\ \widehat{D}^*(z) &= P(L - z\mathbf{1}_N^\top) \end{aligned} \quad \text{for } z \in \mathbb{R}^m.$$

Proof.

$$\begin{aligned} c\mathbf{1}_N^\top + \widehat{D} &= c\mathbf{1}_N^\top + PL \\ &= c\mathbf{1}_N^\top + Pz\mathbf{1}_N^\top + PL - Pz\mathbf{1}_N^\top \\ &= \underbrace{(c + Pz)\mathbf{1}_N^\top}_{c'} + P \underbrace{(L - z\mathbf{1}_N^\top)}_{L'} \\ &= c'\mathbf{1}_N^\top + \widehat{D}' \end{aligned}$$

Therefore, if (c, \widehat{D}) is a solution, then (c', \widehat{D}') is also a solution. The fact the $c = \mathcal{M}(D)$, $\widehat{D} = PL$ is a solution follows from Theorem 5. \square

The fact that the solution produced by the two-stage procedure is not the only solution of (3) is not well known and poses the problem of comparing solutions produced by different methods, *e.g.*, the local optimization methods for the weighted and structured low-rank approximation problems, considered next.

4 Weighted low-rank approximation with centering

For a symmetric positive definite matrix W , we define the weighted norm $\|\cdot\|_W$

$$\|E\|_W := \text{vec}^\top(E)W \text{vec}(E)$$

and consider the weighted low-rank approximation problem (Gabriel and Zamir, 1979; Manton *et al.*, 2003; Srebro, 2004) with centering:

$$\begin{aligned} & \text{minimize} && \text{over } \hat{D} \text{ and } c && \|D - \hat{D} - c\mathbf{1}^\top\|_W \\ & \text{subject to} && \text{rank}(\hat{D}) \leq m. \end{aligned} \quad (12)$$

Contrary to its “unweighted” equivalent of Section 3, now the two-stage procedure of computing the mean in a preprocessing step and then the weighted low-rank approximation of the centered data matrix is suboptimal, as shown in Example 7.

As for the weighted low-rank approximation problem (without centering) there are two basic local optimization solution approaches—alternating projections and variable projections.

Alternating projections algorithm

Using the image representation of the rank constraint

$$\begin{aligned} \text{rank}(\hat{D}) \leq m & \iff \hat{D} = PL, \\ & \text{where } P \in \mathbb{R}^{q \times m} \text{ and } L \in \mathbb{R}^{m \times N}, \end{aligned}$$

we obtain the following problem equivalent to (12)

$$\text{minimize} \quad \text{over } P, L, c \quad \|D - PL - c\mathbf{1}_N^\top\|_W. \quad (13)$$

The method is motivated by the fact that (13) is linear in c and P as well as in c and L . Indeed,

$$\begin{aligned} & \|D - c\mathbf{1}_N^\top - PL\|_W \\ &= \left\| \text{vec}(D) - \begin{bmatrix} I_N \otimes P & \mathbf{1}_N \otimes I_q \end{bmatrix} \begin{bmatrix} \text{vec}(L) \\ c \end{bmatrix} \right\|_W \\ &= \left\| \text{vec}(D) - \begin{bmatrix} L^\top \otimes I_q & \mathbf{1}_N \otimes I_q \end{bmatrix} \begin{bmatrix} \text{vec}(P) \\ c \end{bmatrix} \right\|_W \end{aligned}$$

where \otimes is the Kronecker product $A \otimes B := [a_{ij}B]$. This suggests an iterative algorithm alternating between minimization over c and P with a fixed L and over c and L with a fixed P , see Algorithm 1. Each iteration step is a weighted least squares problem, which can be solved globally and efficiently. The algorithm starts from an initial approximation $c^{(0)}$, $P^{(0)}$, $L^{(0)}$ and on each iteration step updates the parameters with the newly computed values from the last least squares problem. Since on each iteration the cost function value is guaranteed to be non increasing and the

cost function is bounded from below, the sequence of cost function values, generated by the algorithm converges. Moreover, it can be shown (Markovsky and Niranjan, 2008) that the sequence of parameter approximations $c^{(k)}$, $P^{(k)}$, $L^{(k)}$ converges to a locally optimal solution of (13).

Example 7. Figure 2 shows the sequence of the cost function values for a randomly generated weighted rank-1 approximation problem with $q = 3$ variables and $N = 6$ data points. The mean of the data matrix and the approximation of the mean, produced by the Algorithm 1 are, respectively

$$c^{(0)} = \begin{bmatrix} 0.5017 \\ 0.7068 \\ 0.3659 \end{bmatrix} \quad \text{and} \quad \hat{c} = \begin{bmatrix} 0.4365 \\ 0.6738 \\ 0.2964 \end{bmatrix}.$$

The weighted rank-1 approximation of the matrix $D - c^{(0)}\mathbf{1}_N^\top$ has approximation error 0.1484, while the weighted rank-1 approximation of the matrix $D - \hat{c}\mathbf{1}_N^\top$ has approximation error 0.1477. This shows the suboptimality of the two-stage procedure—data centering, followed by weighted low-rank approximation.

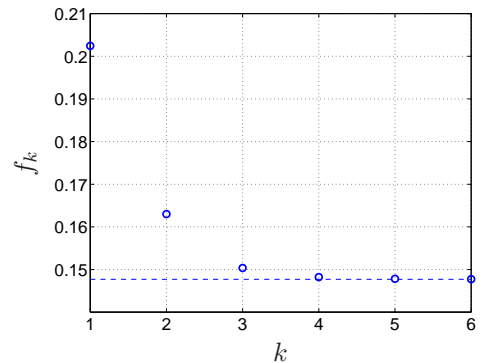


Figure 2: Sequence of cost function values, produced by Algorithm 1.

Variable projections algorithm

The variable projections approach is based on the observation that (13) is a double minimization problem

$$\text{minimize} \quad \text{over } P \in \mathbb{R}^{q \times m} \quad f(P)$$

where the inner minimization is a weighted least squares problem

$$f(P) := \min_{L \in \mathbb{R}^{m \times N}, c \in \mathbb{R}^q} \|D - PL - c\mathbf{1}_N^\top\|_W^2$$

and therefore can be done analytically. This reduces the original problem to a nonlinear least squares problem over P only. We have that

$$f(P) = \text{vec}^\top(D)W\mathbf{P}(\mathbf{P}^\top W\mathbf{P})^{-1}\mathbf{P}^\top W \text{vec}(D),$$

Algorithm 1 Alternating projections algorithm for weighted low-rank approximation with centering.

Input: data matrix $D \in \mathbb{R}^{q \times N}$, rank constraint \mathbf{m} , positive definite weight matrix $W \in \mathbb{R}^{Nq \times Nq}$, and relative convergence tolerance ε .

1: Initial approximation: compute the mean $c^{(0)} := \mathcal{M}(D)$ and the rank- \mathbf{m} approximation $\widehat{D}^{(0)}$ of the centered matrix $D - c^{(0)}\mathbf{1}_N^\top$. Let $P^{(0)} \in \mathbb{R}^{q \times \mathbf{m}}$ and $L^{(0)} \in \mathbb{R}^{\mathbf{m} \times N}$ are full rank matrices, such that $\widehat{D}^{(0)} = \widehat{P}^{(0)}\widehat{L}^{(0)}$.

2: $k := 0$.

3: **repeat**

4: Let $\mathbf{P} := [I_N \otimes P^{(k)} \quad \mathbf{1}_N \otimes I_q]$ and

$$\begin{bmatrix} \text{vec}(L^{(k+1)}) \\ \widehat{c}^{(k+1)} \end{bmatrix} := (\mathbf{P}^\top W \mathbf{P})^{-1} \mathbf{P}^\top W \text{vec}(D).$$

5: Let $\mathbf{L} := [L^{(k+1)\top} \otimes I_q \quad \mathbf{1}_N \otimes I_q]$ and

$$\begin{bmatrix} \text{vec}(P^{(k+1)}) \\ \widehat{c}^{(k+1)} \end{bmatrix} := (\mathbf{L}^\top W \mathbf{L})^{-1} \mathbf{L}^\top W \text{vec}(D).$$

6: Let $D^{(k+1)} := P^{(k+1)}L^{(k+1)}$.

7: $k = k + 1$.

8: **until** $f_k := \|D^{(k)} - D^{(k-1)}\|_W / \|D^{(k)}\|_W < \varepsilon$.

Output: Locally optimal solution $\widehat{c} := \widehat{c}^{(k)}$ and $\widehat{D} = D^{(k)}$ of (13).

where $\mathbf{P} := [I_N \otimes P \quad \mathbf{1}_N \otimes I_q]$. For the outer minimization any standard unconstrained nonlinear (least squares) algorithm can be used.

Example 8. For the same data, initial approximation, and convergence tolerance as in Example 7, the variable projections algorithm, using numerical approximation of the derivatives in combination with quasi-Newton method converges to a locally optimal solution with approximation error 0.1477—the same as the one found by the alternating projections algorithm. The optimal parameters found by the two algorithms are equivalent up to the nonuniqueness of a solution (Theorem 6).

5 Hankel low-rank approximation with centering

Unstructured low-rank approximation is a tool for data modeling by linear static models. Similarly, Hankel structured low-rank approximation is a tool for data modeling by linear time-invariant dynamic models. In the static case, the ordering of the data points is inessential. In contrast, in the dynamic case, the index represents time (the data is a time series) and therefore the ordering is important. In order to emphasise this fact and make link with the system theory and signal

processing literature, where linear time-invariant dynamic systems are used, in this section, we denote the data as follows:

$$\mathcal{D} = (d(1), \dots, d(N)), \quad d(t) \in \mathbb{R}^q.$$

Also, we define the data vector

$$d = [d^\top(1) \quad \dots \quad d^\top(N)]^\top \in \mathbb{R}^{qN}.$$

A linear time-invariant dynamic model can be represented by a difference equation

$$R_0 d(t) + R_1 d(t+1) + \dots + R_n d(t+n) = 0, \quad (14)$$

where R_0, R_1, \dots, R_n are parameters of the model and the integer n is called the lag of the equation. With some loss of generality we will assume that the highest power coefficient R_n is of full row rank. Then, the number of rows \mathbf{m} of R_n is equal to the number of inputs in an input/output representation of the model. The pair of numbers (\mathbf{m}, n) specifies the complexity of the model.

A Hankel structured matrix is a matrix with equal anti-diagonals

$$\mathcal{H}_{n+1}(d) := \begin{bmatrix} d(1) & d(2) & \dots & d(N-n) \\ d(2) & d(3) & \dots & d(N-n+1) \\ \vdots & \vdots & \ddots & \vdots \\ d(n+1) & d(n+2) & \dots & d(N) \end{bmatrix}.$$

If the data d satisfies the difference equation (14), then

$$\text{rank}(\mathcal{H}_{n+1}(d)) \leq \underbrace{(n+1)\mathbf{m} + n}_{=:r} < (n+1)q,$$

i.e., $\mathcal{H}_{n+1}(d)$ is rank deficient. This fact is used to define the following identification problem

$$\begin{aligned} & \text{minimize} \quad \text{over } \widehat{d} \quad \|d - \widehat{d}\|_2 \\ & \text{subject to} \quad \text{rank}(\mathcal{H}(\widehat{d})) \leq r \end{aligned} \quad (15)$$

which is also a low-rank approximation problem, with the extra constraint that the approximation matrix should be Hankel structured. In the case of data centering we have the following modified Hankel low-rank approximation problem:

$$\begin{aligned} & \text{minimize} \quad \text{over } \widehat{d} \text{ and } c \quad \|d - \mathbf{1}_N \otimes c - \widehat{d}\|_2 \\ & \text{subject to} \quad \text{rank}(\mathcal{H}(\widehat{d})) \leq r. \end{aligned} \quad (16)$$

Algorithm

Consider the kernel representation of the rank constraint

$$\begin{aligned} & \text{rank}(\mathcal{H}(\widehat{d})) \leq r \\ \iff & \text{there is full rank matrix } R \in \mathbb{R}^{\mathbf{p} \times (n+1)q} \\ & \text{such that } R\mathcal{H}(\widehat{d}) = 0. \end{aligned}$$

We have

$$RH(\widehat{d}) = 0 \iff T(R)\widehat{d} = 0,$$

where

$$T(R) = \begin{bmatrix} R_0 & R_1 & \cdots & R_n & & & \\ & R_0 & R_1 & \cdots & R_n & & \\ & & \ddots & \ddots & & \ddots & \\ & & & R_0 & R_1 & \cdots & R_n \end{bmatrix}$$

(all missing elements are zeros). Let M be a full rank matrix, such that

$$\text{image}(M) = \ker(T(R)).$$

Then the constraint of (16) can be replaced by

$$\text{there is } \ell, \text{ such that } \widehat{d} = M\ell,$$

which leads to the following problem equivalent to (3)

$$\text{minimize over } R \quad f(R),$$

where

$$f(R) := \min_{c, \ell} \left\| d - \begin{bmatrix} \mathbf{1}_N \otimes I_q & M \end{bmatrix} \begin{bmatrix} c \\ \ell \end{bmatrix} \right\|.$$

The latter is a standard least-squares problem, so that the evaluation of f for a given R can be done efficiently. Moreover, one can exploit the special structure of the Toeplitz matrix T in the computation of M and in the solution of the least-squares problem.

Example 9. The data sequence is

$$d(t) = 0.9^t + 1, \quad t = 1, \dots, 10.$$

The sequence $(0.9^1, \dots, 0.9^{10})$ satisfies a difference equation (14) with lag $n = 1$ (a first order autonomous linear time-invariant model), however, a shifted sequence $d(t) = 0.9^t + c$, with $c \neq 0$, does not satisfy such an equation. The mean of the data is $\mathcal{M}(D) = 1.5862$, so that the centered data $d(t) - \mathcal{M}(D)$ does not satisfy (14) with lag $n = 1$. Solving the Hankel structured low-rank approximation problem with centering (15), however, yields the exact solution $\widehat{c} = 1$.

Preprocessing by centering the data is common in system identification (Ljung, 1999). Example 9 shows that preprocessing can lead to suboptimal results. Therefore, there is need for methods that combine data preprocessing with the existing identification methods. The algorithm derived in this section is such a method for identification in the errors-in-variables setting. It can be modified for output error identification, *i.e.*, assuming that the input of the system is known exactly, however, this will be pursued elsewhere.

6 Conclusions

Simultaneous low-rank approximation with centering was considered and it was proven that in the unweighted case the problem can be solved by first computing the centering parameter and then the low-rank approximation of the centered matrix with that parameter. Similar two-stage procedure, however, is suboptimal in the cases of weighted and structured low-rank approximation problems, where one has to simultaneously optimize over the centering parameter and the low-rank approximant. Two algorithms, one based on alternating projections and one based on variable projections, were developed for the latter cases.

Acknowledgments

I would like to thank Guido Sanguinetti for comments on the paper. Research supported by PinView (Personal Information Navigator adapting through VIEWing), an EU FP7 funded Collaborative Project 216529.

References

- Bishop, C. (2006). *Pattern recognition and machine learning*. Springer.
- Gabriel, K. and S. Zamir (1979). Lower rank approximation of matrices by least squares with any choice of weights. *Technometrics* **21**, 489–498.
- Gleser, L. (1981). Estimation in a multivariate "errors in variables" regression model: large sample results. *The Annals of Statistics* **9**(1), 24–44.
- Jolliffe, I. (2002). *Principal component analysis*. Springer-Verlag.
- Ljung, L. (1999). *System Identification: Theory for the User*. Prentice-Hall. Upper Saddle River, NJ.
- Manton, J., R. Mahony and Y. Hua (2003). The geometry of weighted low-rank approximations. *IEEE Trans. Signal Process.* **51**(2), 500–514.
- Markovsky, I. and M. Niranjan (2008). Approximate low-rank factorization with structured factors. *Comput. Statist. Data Anal.*
- Markovsky, I., J. C. Willems, S. Van Huffel and B. De Moor (2006). *Exact and Approximate Modeling of Linear Systems: A Behavioral Approach*. SIAM.
- Sanguinetti, Guido, Marta Milo, Magnus Rattray and Neil Lawrence (2005). Accounting for probe-level noise in principal component analysis of microarray data. *Bioinformatics* **21**, 3748–3754.

- Srebro, N. (2004). Learning with Matrix Factorizations. PhD thesis. MIT.
- Wentzell, P., D. Andrews, D. Hamilton, K. Faber and B. Kowalski (1997). Maximum likelihood principal component analysis. *J. Chemometrics* **11**, 339–366.
- Zhang, Z. and H. Zha (2005). Principal manifolds and nonlinear dimension reduction via local tangent space alignment. *SIAM Journal on Scientific Computing* **26**, 313–338.