Determination of optimal prediction oriented multivariate latent factor models using loss functions

C. Weihs [*]
Fachbereich Statistik
Universität Dortmund

T. Hothorn
Institut für Medizininformatik, Biometrie und Epidemiologie
Friedrich-Alexander-Universität Erlangen-Nürnberg

February 2002

Abstract

In this paper a projection pursuit method is developed which determines optimal multivariate latent factor models based on a flexible loss function. This way, the unknown model coefficients are estimated with respect to optimal predictive power. The specification of the loss function in practical applications is discussed. The method is illustrated by means of simulation examples.

Keywords
latent factor models, predictive power, loss functions, projection pursuit

1 Introduction

The target of this paper is to construct the minimum number of latent variables corresponding to a specified loss function which are sufficient to describe the dependency of certain response variables on certain possibly influencing factors. The basic ideas are two-fold. On the one hand side the paper is based on ideas for estimating so-called reduced rank models in which latent variables are searched for in both response variables and influencing factors (cp. Schmidli, 1995, and for a more recent overview Reinsel and Velu, 1998). These ideas are applied to the construction of best prediction oriented models in which only latent factors are assumed. This is realized, on the other hand, by constructing best orthogonal latent variables corresponding to certain prediction oriented loss functions by means of simulated annealing, a general search method for function optimization (cp. e.g. Press et al., 1992a, and Röhl et.al., 2002) implemented to find the best loadings matrix with optimal rank and orthogonal scores.

[*] e-mail: weihs@statistik.uni-dortmund.de

The paper is organized as follows. In section 2 the models to be dealt with will be introduced. Section 3 presents the basic ideas for constructing prediction optimal latent factor models. Section 4 derives the criterion for the best prediction oriented loadings matrix with orthogonal scores. In section 5 the side condition of orthogonality of scores is discussed. Sections 6 and 7 present the optimization algorithm, and section 8 some simulation results. In section 9 the paper is concluded.

2 Predictions with multivariate linear models

Particularly in the case of **more than one response variable** methods for **optimal prediction** with **multivariate multiple models** are of interest. The various prediction methods, on the one hand side, differ in the optimality criterion judging the predictive power of a model. On the other side, the methods differ with respect to various restrictions on the models taken into account. In **principal components regression**, e.g., only linear models with principal components as regressors are taken into account. In general, multivariate multiple linear models are defined as follows.

> **Multivariate multiple linear model**
> Let Y the data matrix of the m response variables and X the data matrix of the K influential factors, both with n observations with respect to the same objects. The multivariate multiple linear model has then the form :
> $Y = XA + E$, where
> A is a matrix of unknown coefficients, and
> E is the matrix of model errors.

If it is assumed that the **structure of the matrix of coefficients is not restricted** and that the model errors are iid and normal, then the **least squares estimator** has the desirable property that each response can be treated individually estimating the coefficients. Thus, we can restrict ourselves to multiple linear models for one response variable.

Moreover, this argument uncovers the most important reasons for multivariate prediction in linear models:

- **dependent model errors:** the model errors of the different response variables are dependent, e.g. when the responses were measured in the same experiment or at the same object,
- **dependent coefficients:** the model coefficients of the different responses are related to each other, e.g. when some of them are restricted to be equal or when the matrix of coefficients does not have maximum rank, or is restricted by a side condition.

In what follows, we exclusively consider an especially important multivariate model with a coefficients matrix with reduced rank. It is assumed that the expectation of the responses is linearly dependent on (a small number of) so-called latent (or implicit) variables, which are themselves linear combinations of the original factors. This leads to the so-called latent factor model, in which the coefficients matrix is of a special form.

Multivariate multiple linear latent factor model
Let Y the data matrix of m (mean centered) responses and X the data matrix of K (mean centered) influential factors, each with n observations at the same objects. A **multivariate multiple linear latent factor model** has then the form:
$Y = ZB + E = XGB + E$, where
$Z := XG$ is an $n \times r$-matrix of the scores of r latent factors with $r < K$ and $Z'Z = I$,
G, B are $K \times r$ - and $r \times m$ - matrices of unknown coefficients, respectively, and
E is the matrix of model errors.
The matrix G is called the matrix of **factor loadings.**

Note that these latent factor models should not be mixed with the reduced rank models also well-known from the literature where the matrix product GB is a full-rank decomposition of the overall coefficients matrix with G and B both having rank r (cp. e.g. Schmidli, 1995, p. 55). This model type is generated from, e.g., canonical correlation analysis and redundancy analysis (cp. e.g. Schmidli, 1995, p. 61-64). Note that the rank restriction on the coefficient matrix B in reduced rank models implies that $r \leq m$, i.e. that the number of latent factors relevant to the responses Y is not greater than the number of responses. Instead, our models are asymmetrical in that only latent factors are assumed, and m might well be $< r$. This way, e.g., principal components regression models are enclosed even for the case, e.g., of one response and two principal components.

3 Predictions with latent factor models

In the following, we first introduce a theoretical measure discussed for predictive power for **reduced rank models** (cp. Schmidli, 1995), which is then applied to our latent factor models. This measure is based on the definition of simultaneous predictions in p points, given here for latent factor models. In this definition it is assumed that in the latent factor model there is no restriction on the matrix of coefficients B of the latent factors Z so that least squares estimation is adequate. Note that this, in particular, makes the difference to reduced rank models where the matrix B has to have the same rank as the loadings matrix G.

Point prediction for latent factor models
For a known fixed $p \times K$-matrix of influential factors $X_0 = (x_{10} \dots x_{p0})'$ the **point prediction** for a corresponding realization $Y_0 = (y_{10} \dots y_{p0})'$ of the $p \times m$-matrix of random variables $Y_1, \dots, Y_m$ based on training data $X = (x_1 \dots x_n)'$ and $Y = (y_1 \dots y_n)'$ is defined by the expectation of Y_0 based on least-squares estimations of the unknown model coefficients B with respect to X and Y :
$\hat{Y}_0 := Z_0\hat{B} = X_0(G\hat{B}) =: X_0\hat{A}$, where $\hat{B} := Z'Y = (XG)'Y$.

Note that the least squares estimates of the coefficients B have the above simple form since the score matrix Z is assumed to be orthogonal, i.e. $Z'Z = I$. The following class of measures for the predictive power is based on a fairly general class of loss functions.

Measures for predictive power
A **loss function** for the assessment of predictive power is defined as:
$V_\Gamma(Y_0) = \frac{1}{p}||(Y_0 - \hat{Y}_0)\Gamma^{-0.5}||_F^2 := \frac{1}{p}trace((Y_0 - \hat{Y}_0)'(Y_0 - \hat{Y}_0)\Gamma^{-1})$, where Γ is a deliberately specifiable, but fixed $m \times m$ weight matrix. $||A||_F$ is called Frobenius-norm of the matrix A.
For the mean assessment of predictive power of a prediction model often the **expected loss** (the so-called **risk**) is used. Note that the (conditional) expectation is taken both with respect to Y_0 given X_0, and with respect to Y given X, where the responses given the factors are assumed to be independently identically distributed. This expected loss is often called Mean Squared Expected Prediction error (MSEP):
MSEP $:= \frac{1}{m}E_{Y|X}E_{Y_0|X_0}||(Y_0 - \hat{Y}_0)\Gamma^{-\frac{1}{2}}||_F^2$.

Obviously, these loss functions are minimal, when the point predictions are coincident with the realizations of the response variables in all relevant prediction situations. Note that the MSEP is defined as a conditional expectation, conditional on X and X_0.

One problem of the predictive power measure MSEP is that the distribution of the responses most of time is at least partially unknown. In such cases, usually one of the following three methods is used:

- the **fit method:** one chooses $Y_0 = Y$ (and $X_0 = X$)
- the **asymptotic method:** one tries to develop asymptotic approximations for MSEP;
- the **resampling method:** one uses resampling algorithms such as cross validation or bootstrapping for the approximation of MSEP. This method is explained in more detail in section 6.

By different choices of the **weight matrix** Γ one obtains different model selection criteria. One possible choice for Γ is the covariance matrix of the measurement errors of the responses. A motivation for such a choice would be, that the influential factors are assumed fixed (conditional considerations), and, thus, the point predictions differ only by the measurement errors from the realizations of the responses.

In some special cases the form of the covariance matrix of measurement errors can be derived from the type of the performed experiment. The structure of the weight matrix Γ is, thus, determined by the data structure of the studied problem (cp. Schmidli, 1995, p. 59).

1. **Covariance matrix** Σ_0 of the measurement errors **known**:
   If the responses are measured in a routine experiment, sometimes one can deduce estimates of the size of the measurement errors from repeated former experiments. If such estimates are exact enough, they can be used to derive reliable estimates of the error covariance matrix.

2. **Covariance matrix proportional to identity matrix I:**
   If all responses are measured by means of the same measurement procedure, it is often plausibel to assume that all measurement errors are of the

same size. If additionally the measurements of different responses are obtained in different experiments, then the measurement errors of different responses can be assumed independent. Altogether, this leads to an error covariance matrix proportional to the identity matrix.

3. **Covariance matrix diagonal:** If the responses are measured in different experiments, but with different measurement procedures, then a diagonal error covariance matrix appears plausibel which is not necessarily proportional to the identity matrix.

4. **Covariance matrix unrestricted positive definite:** If the responses are measured in the same experiment, then the measurement errors are correlated in general. If no further information about the measurement errors is available, then one tends to assume invertibility at least.

4 Estimation of the loadings matrix

Since $\hat{B} := Z'Y = (XG)'Y = G'X'Y$ was assumed to be the least squares estimate, only the loadings matrix G is to be estimated. In order to optimally estimate the matrix G, the predictive power measure in section 3 is used.

We will start, though, with the **fit method** introduced above. Thus, we will replace X_0, Y_0 by X, Y. Then, we obtain the following loss function to assess a loadings matrix G:

$$\begin{aligned} V_\Gamma(Y): &= \frac{1}{p}\|(Y - XGG'X'Y)\Gamma^{-0.5}\|_F^2 \\ &= \frac{1}{p} trace((Y - XGG'X'Y)\Gamma^{-1}(Y - XGG'X'Y)'). \end{aligned}$$

In special cases, from this loss function one obtains well-known criteria for the **estimation** of G for fixed rank r which lead to **analytical solutions** when normal errors and $r \le min(m, K)$ are assumed (cp. Schmidli, 1995, pp. 58-63).

(2) In the case **'Covariance matrix proportional to the identity matrix I'** one obtains the so-called **redundancy analysis RDA**. An equivalent formulation of the corresponding optimization problem reads: $trace(XG)'(YC) = max!$, where $(XG)'(XG) = I$ and $C'C = I$.

(4) In the case **'Covariance matrix unrestricted positive definite'** one obtains the so-called **canonical correlation analysis CCA**. An equivalent formulation of the corresponding optimization problem reads: $trace(XG)'(YC) = max!$, where $(XG)'(XG) = I$ and $(YC)'(YC) = I$.

Let us now turn to oblique prediction criteria. For the **determination of the optimal rank** r in the literature it is proposed to use the MSEP criterion in the following way (Schmidli, 1995, p. 87):

$$\text{MSEP}_\Gamma(r) := \frac{1}{p} E_{Y|X} E_{Y_0|X_0} \|(Y_0 - X_0\hat{G}_r(Y)\hat{G}_r(Y)'X'Y)\Gamma^{-0.5}\|_F^2$$

where the observations are sequentially preliminarily eliminated from the data set to be utilized as $(Y0, X0)$, and only the rest of the observations is used to produce the estimate $\hat{G}_r(Y)$ of the matrix G_r (cross validation). Thus, $\hat{G}_r(Y)$ is exclusively based on Y so that implicitly Y_0 is set to Y in the criterion and one of the above cited analytical solution methods might be used. Then, the rank r is determined by minimizing $\mathrm{MSEP}_\Gamma(r)$. In this way, a method for the construction of an approximately optimal model corresponding to the prediction criterion can quite easily be employed.

In this paper, though, we propose a more general procedure, namely **to estimate** the matrix $\hat{G}_r$ **directly by means of the general prediction criterion**, i.e. by minimizing the general MSEP criterion avoiding to use $Y_0 = Y$ anywhere. This way, a much more suitable model can be built in that the prediction generated by the constructed model is guaranteed to be optimal. The **loss function for the assessment of the predictive power** of a matrix G_r of rank r can be written as follows:

$$V_\Gamma(G_r) = \frac{1}{p}||(Y_0 - X_0 G_r G_r' X'Y)\Gamma^{-\frac{1}{2}}||_F^2 \quad \text{,where} \quad (XG_r)'(XG_r) = Ir$$

The corresponding expected loss has then the form:

$$MSEP_\Gamma(r) := \frac{1}{p} E_{Y|X} E_{Y_0|X_0} ||(Y_0 - X_0 G_r G_r' X'Y)\Gamma^{-0.5}||_F^2, \qquad (1)$$
$$\text{where} \quad (XG_r)'(XG_r) = I_r$$

Here, **the whole matrix G_r is to be optimized** over all possible ranks r. The specification of the weight matrix Γ should not be restricted. The matrix G_r projects the original data X into an r-dimensional subspace of the original K-dimensional space. What we thus look for, is an **optimal projection.**

5 Feasible loadings matrices

As a preparation for the construction of general test problems for the above minimum-norm problem (1) with side condition we determine the general solution of the side condition for fixed rank r. In passing we show that the value of the loss function stays equal in certain solution classes.

For the determination of a solution G_r of the side condition $(XG_r)'(XG_r) = I_r$ we exploit the singular value decomposition of the matrix $(X'X)$:

$$X'X = V\Lambda V'.$$

Let V_r the matrix with the first r columns of the orthonormal matrix V and Λ_{rr} be the diagonal matrix with the first r rows and columns of the diagnonal matrix Λ.
Then, a first solution of the side condition is $G_r := V_r \Lambda_{rr}^{-0.5}$ since

$$\begin{aligned}(XG_r)'(XG_r) &= G_r' X' X G_r = G_r' V \Lambda V' G_r = \Lambda_{rr}^{-0.5} V_r' V \Lambda V V_r \Lambda_{rr}^{-0.5} \\ &= \Lambda_{rr}^{-0.5} (I_r 0) \Lambda (I_r 0)' \Lambda_{rr}^{-0.5} = \Lambda_{rr}^{-0.5} \Lambda_{rr} \Lambda_{rr}^{-0.5} = I_r\end{aligned}$$

Note that Λ_{rr} is assumed to be invertible. In order that this is true for all $r \leq K$, we assume for the moment that X has maximum column rank, i.e. that $rank(X) = K$.

Thus, the (normalized) first r **principal components** V_r are one solution of the side condition of the optimization problem. This idea can, however, be easily extended to a more general class of solutions.

Let $U := VW$, where W is assumed to have maximum column rank, i.e. $rank(W) = K$. Then let $G_r := U_r(W'\Lambda W)_{rr}^{-0.5} = (VW_r)(W_r'\Lambda W_r)^{-0.5}$, where W_r is the matrix with the first r columns of W, and $(W'\Lambda W)_{rr}$ the matrix with the first r rows and columns of the matrix $(W'\Lambda W)$. This implies:

$$\begin{aligned}(XG_r)'(XG_r) &= G_r'X'XG_r = G_r'V\Lambda V'G_r \\ &= ((W_r'\Lambda W_r)^{-0.5})'W_r'V'(V\Lambda V')VW_r(W_r'\Lambda W_r)^{-0.5} \\ &= ((W_r'\Lambda W_r)^{-0.5})'(W_r'\Lambda W_r)(W_r'\Lambda W_r)^{-0.5} \\ &= I_r.\end{aligned}$$

Note that $rank(X) = K$ together with $rank(W) = K$ guarantees that $(W_r'\Lambda W_r)^{-1}$ exists for all $r \leq K$.

Examples for such W are so-called **permutation matrices**, which interchange the ordering of the columns of a matrix when multiplied from the right. In this way, U_r can contain any r columns of V. Note that this corresponds to the standard practice of **principal component regression** to choose the best predicting principal components, and not just the first r components as predictors. But note moreover that more general W are allowed, too.

Let us now consider the case of a **rank deficient matrix** X. In this case there are additive transformations of a feasible loadings matrix G_r not only leading to feasible loadings matrices again but also to **equivalent loadings matrices**, i.e. to the same utility function value. In such cases, there exists a whole manifold of matrices F_r equivalent to some feasible G_r. Indeed, assume that $X = X_bC$ is a **full rank decomposition** of X, i.e. X_b has maximum column rank $R \leq K$ so that the columns of X_b build a basis of the range of X, and C has maximum row rank. Then let H_r any $r(\leq R)$ dimensional solution of
$CH_r = 0$, i.e. let the columns of H_r be any elements of the null space of C. Then
$F_r = G_r + H_r$ has the property
$CF_r = C(G_r + H_r) = CG_r$, which implies $XF_r = X_bCF_r = X_bCG_r = XG_r$. Thus,
$(XF_r)'(XF_r) = (XG_r)'(XG_r) = I_r$, i.e. feasibility is retained, and

$$\begin{aligned}V_\Gamma(F_r) &= \frac{1}{p}||(Y_0 - (X_0F_r)(XF_r)'Y)\Gamma^{-0.5}||_F^2 \\ &= \frac{1}{p}||(Y_0 - (X_0G_r)(XG_r)'Y)\Gamma^{-0.5}||_F^2 \\ &= V_\Gamma(G_r),\end{aligned}$$

i.e. the loss is equal for G_r and all F_r.

Example: Let $K = 2, r = 1$, and $X = x_bc'$, where x_b is a basis (column) vector of the range of X, and c' is a row vector. Then for any given loadings vector g_1 the vectors $f_1 = g_1 + h_1$ with $c'h_1 = 0$ are equivalent loadings vectors. Thus, the equivalent vectors are lying on a line in $I\!R^2$.

Independent of the rank of X there are other equivalent loadings matrices. Indeed, if an r dimensional subspace is fixed by means of G_r, then it is easy to show that the value of the above loss function stays constant when this solution G_r is transformed by means of an r dimensional orthonormal transformation

H_r, an $r \times r$ matrix. Let $F_r = G_r H_r$, $H_r' H_r = I_r = H_r H_r'$, then

$$\begin{aligned}
V_\Gamma(F_r) &= \frac{1}{p} \|(Y_0 - X_0 F_r F_r' X'Y)\Gamma^{-0.5}\|_F^2 \\
&= \frac{1}{p} \|(Y_0 - X_0 G_r H_r (G_r H_r)' X'Y)\Gamma^{-0.5}\|_F^2 \\
&= \frac{1}{p} \|(Y_0 - X_0 G_r H_r H_r' G_r' X'Y)\Gamma^{-0.5}\|_F^2 \\
&= \frac{1}{p} \|(Y_0 - X_0 G_r G_r' X'Y)\Gamma^{-0.5}\|_F^2 \\
&= V_\Gamma(G_r)
\end{aligned}$$

Obviously, this property is even valid independent of the validity of the side condition for the matrix G_r! Thus, **by means of r dimensional orthonormal transformations equivalence classes of loadings matrices G_r are defined** in any case.

Note that the existence of equivalence classes leads to problems with **reproducibility**, i.e. in simulation experiments one generally should not expect to reproduce the matrix G_r used for data generation by means of the solver of the mininum-norm problem. On the other hand, models with matrices X **with non-deficient rank and no model errors** can be used to check algorithms constructed to determine the optimal loadings matrix G. Indeed, for such models it is clear that the minimal model error is zero even in subsamples, since by construction it is valid that
$Y = XGB$ with $(XG)'(XG) = I$.
In subsamples X_s the restriction $(X_s G)'(X_s G) = I$ might not be valid, though. However, if $X_s G$ has maximum column rank, $X_s G$ can be orthonormalized by an invertible matrix T. Then,
$(X_s GT)'(X_s GT) = I$ and
$Y_s = X_s GB = X_s GTT^{-1}B = (X_s GT)B_s$ with $B_s = T^{-1}B$ as the new coefficient matrix.

Thus, an algorithm to determine the optimal loadings matrix should find a matrix with (at least nearly) zero model error.

Example: In the case $K = 2, r = 1$, G has only one column, and the matrix T is a scalar $\neq 0$. Therefore, the angle between the optimal loadings vector and the line through the vector G is zero.
Note that the found equivalence classes neither depend upon the weight matrix Γ, nor on the values of the responses in Y or Y_0, whereas the value of the minimum-norm criterion, and thus the optimum, very well depends on these matrices.

6 Projection pursuit

In order to solve the minimum-norm problem under the side condition, we propose a search method constructed to find the optimal projection. Such methods are often called projection pursuit methods. Basically, the method looks as follows:

1. Generate a random subsample (X, Y) from the observations.

2. Mean centering of X, subtract the mean of X, also from X_0.
3. As the MSEP is a conditional expectation it is necessary in a simulation study to generate different values for Y and Y_0 with the same partition in X and X_0 to estimate the MSEP correctly - when G is not linear function of X, Y. This is done by adding different values of E – according to the error structure – to generate Y and Y_0. With real data this can be done by a bootstrap method.
4. Mean centering of Y, subtract the mean of Y, also from Y_0.
5. Determine for each rank r the minimum-norm optimal G_r for this sample, where it is guaranteed that $(XG_r)'(XG_r) = I_r$.
6. Repeat step 3-5 often.
7. Repeat the first six steps sufficiently often.
8. Estimate the expected loss by means of the mean over all samples.
9. Choose the optimal G_r, i.e. the optimal rank r.

One candidate for the optimization method to be used in the second step is simulated annealing (Bohachevsky et al., 1986). The problem with this method, and with every search method, is to guarantee the side condition. A feasible starting matrix is known from section 5, namely $G_r := V_r\Lambda_{rr}^{-0.5}$. Also based on section 5, at the moment we restrict ourselves to problems of the optimal choice of a maximum column rank matrix W which transforms the eigenvector matrix V. In order to guarantee maximum column rank we even restrict ourselves to orthonormal transformation matrices W. Then, the side condition is automatically fulfilled. Thus, projection pursuit has to look for that W that minimizes the expected loss (1). If in a search step the generated transformation matrix W is not orthonormal, the Gram-Schmidt method can be used for post-orthonormalisation. The success of a step is judged only after such an orthonormalisation. Based on these ideas the optimization algorithm can be described as follows.

7 The optimization algorithm

We now discuss the optimization algorithm for the entries in the transformation matrix W of the loadings matrix $G_r := V_r\Lambda_{rr}^{-0.5}$ to minimize the expected loss (1) for fixed rank r in detail. Simulated annealing is used for this task since it does not need derivatives and since it has proved to overcome local minima.

In physics it is well-known that freezing and crystallizing of liquids overcomes local energy minima. This strategy serves as the prototype for a computer program. To model the natural procedure, we need a configuration space (a discrete or continuous domain), a mechanism which describes how to get from one configuration to another and a cooling schedule describing how to decrease the temperature $T \quad (T_0 \to T_1 \to \ldots \to T_n \to \ldots)$. In a concrete optimization, the temperature T is not a physical quantity but an abstract parameter which controls the optimization.

In our problem, the configuration space is $I\!R^{K^2}$, the space of vectorized transformation matrices w. In our algorithm, the cooling schedule is a simple linear scheme, and at each parameter value T a markov chain based on a stochastic version of the well-known Nelder/Mead search algorithm (see below) serves as the transition mechanism between succeeding configurations. At each parameter value T – beginning at any configuration w_0 – we start a markov chain. This chain generates random realizations (after some burn-in period) from a density proportional to $\pi(w) = exp(-V_\Gamma(W)/T)$, where the loss function V_Γ is expressed as a function of the transformation matrix W corresponding to the vector w.

A trial point w_p is chosen according to the stochastic Nelder/Mead transition function $q(w_0, w_p)$. The efficiency of the optimization algorithm depends on this transition function and on the cooling schedule. The transition function 'explores' the configuration space. Information about this space enhances the search. The trial point is accepted with probability $\pi(w_p)/\pi(w_0) = exp(-(V_\Gamma(W_p) - V_\Gamma(W_0))/T))$. In this way, in our problem transformations leading to a decrease of loss are accepted in any case, but also transformations increasing the loss are accepted with some probability. This is the reason, why simulated annealing is able to overcome local optima and thus avoids the selection of multiple starting points.

After a number of steps in the markov chain, the parameter T_n will be decreased by the simple linear scheme $T_{n+1} = \alpha T_n, 0 < \alpha < 1$, and a new chain will be created (the starting point of the new chain is the end point of the last one).

We use an implementation of the simulated annealing algorithm based on a routine in 'Numerical Recipes in C' (Press et al., 1992a). In this routine the transition function is a stochastic version of the search algorithm of Nelder and Mead (Press et al., 1992b). This algorithm encloses the optimum by shrinking simplices. The shrinking is proportional to the parameter T_n. Therefore, as T_n approaches zero, the allowed movements will be more and more local, and the algorithm converges to the next optimum. In this respect, the cooling schedule 'encodes' the size of the neighborhood that can be visited from a point of the markov chain. Because of the bigger values of the parameter T in the beginning of the procedure there is a good chance that this optimum is a global one. This algorithm had to be somewhat extended because in each search step the generated loadings matrix needs to be orthonormal. Therefore, the Gram-Schmidt method is used for post-orthonormalisation of W. The success of a step is judged only after this orthonormalisation.

For the not yet specified parameters of the simulated annealing procedure we have chosen the following values: $T_0 = 0.5$ (section 8.1), and 0.01 (section 8.2). $\alpha = 0.8$ and the number of iterations in the markov chain at each temperature is 100. The minimum temperature is 1E−07.

8 Simulation

8.1 Rank check

We performed the following simulation experiment to check whether our algorithm finds the correct rank of G for the identity as the weight matrix Γ.

1. Generate a design matrix $X_{all} = (X' \, X_0')'$: We use a 100×7 matrix with independent realizations of a binomial ($\mathbf{B}(10, 0.5)$) distribution. Note that it is not the intention to generate random design matrices here, but to use a simple method to create simulation datasets. X_{all} should be interpreted as fixed.

2. Mean centering of X_{all}.

3. Generate the true loadings matrix G_r , $r = 3$ (cp. section 5): In simulations 1 $G_r := V_r \Lambda_{rr}^{-0.5}$ is used, whereas in simulations 2 we use $G_r := (VW_r)((W^{-1})_{(r)} \Lambda W_r)^{-0.5}$, where W is a realization of a random matrix equal to the right singular vectors of a matrix with independent binomial ($I\!B(10, 0.2)$) entries. Again, the intention of this choice of the fixed 'true' W was technical.

4. Generate a matrix of the response variables: We generate a 100×2 matrix $Y_{all} = X_{all} G_r B$ where the $r \times p$ matrix B is filled columnwise with the entries $1, \ldots, rp = 3 \cdot 2 = 6$. Note that $r = 3$ is not smaller than $min(m, K)$ so that we are not dealing with a reduced rank model but only with what we call a latent factor model. Generate the learning and test samples: We use random 50% splits.

5. Mean centering of subsample X, use the estimate of the mean of X to center X_0.

6. We use two variants of the simulations, 1 and 2, respectively, namely without error E and with an error matrix E with independent realizations of a 2-dimensional normal distribution with expectation zero and a diagonal covariance matrix with diagonal entries 0.1 and 0.001, respectivly. Note that the maximum error is a factor 30 smaller than the maximum response value. Thus, errors are small.

7. Mean centering of subsample Y, use the estimate of the mean of Y to center Y_0.

8. Repeat steps 6-8 five times.

9. Apply the optimization algorithm to generate the optimal loadings matrix for ranks $1, 2, \ldots, K-1$. Note that in simulations 1 for $r = 3$ the iteration should (at least in principle) stop in the starting point, if the structure of the subsample would be near to the structure of the whole sample.

10. Repeat steps 4-9 twenty times

11. From these 100 replications estimate the mean loss function for the different ranks.

12. Determine the best rank based on the average losses.

The results of this simulation experiment are very promising. The optimum rank was between 3 and 6, but the difference between the loss 3 till 6 is more or less negliable – except for simulation 2 with no errors.

Table 1: Mean loss for ranks 1-6

sim	var	1	2	3	4	5	6
1	0	6.47E−03	3.79E−05	9.94E−08	1.61E−08	2.34E−08	1.10E−08
1	0.001	8.17E−03	1.99E−03	1.94E−03	1.96E−03	2.01E−03	2.05E−03
1	0.1	2.01E−01	1.89E−01	1.89E−01	1.89E−01	1.89E−01	2.02E−01
2	0	6.51E−03	4.06E−04	1.82E−06	2.31E−08	2.35E−08	1.27E−08
2	0.001	9.91E−03	2.20E−03	1.93E−03	1.95E−03	1.99E−03	2.05E−03
2	0.1	2.06E−01	1.89E−01	1.89E−01	1.89E−01	1.89E−01	2.00E−01

8.2 Loadings check

In order to check whether the correct loadings are generated by the algorithm we simulated a model with $K = 2, r = 1$, and a matrix X with non-deficient rank $(= 2)$. In this case the optimal loss is zero and the angle of the optimal loadings vector to the line through the vector G used for the generation of the observations of Y is zero, too (cp. section 5). To check this, we performed the following simulation experiment again using the identity as the weight matrix

1. Generate a design matrix X_{all}: We use a 300×2 matrix with independent realizations of a binomial $(\mathbf{B}(10, 0.5))$ distribution in both columns.
2. Mean centering of X_{all}.
3. Generate the true loadings matrix $G_r, r = 1$, as $G_r := V_r \Lambda_{rr}^{-0.5}$.
4. Generate a vector of the response variables: We generate a 100-vector $Y_{all} = X_{all} G_r B$, where the scalar B is equal to 1 .
5. Generate the learning and test samples: We use random 50% splits.
6. Mean centering of subsample X and Y, use the estimate of the mean of X to center X_0 and the mean of Y to center Y_0 respectively.
7. Apply the optimization algorithm to generate the optimal loadings matrix for rank 1.
8. Repeat steps 4 and 5 one hundert times.
9. Determine the maximum deviation of the found optimal losses from zero.
10. Determine the distribution of the angles of the found optimal loadings vectors to the line through the true loadings vector G_r.

As there is no error in this model it is not necessary to generate different Y and Y_0.

The results are very promising, too, since the maximum loss is $2.5E-28$, and the maximum angle is $1.2E-06$.

9 Conclusion

In this paper we developed a projection pursuit procedure for the optimization of a flexible prediction oriented loss function to estimate a latent factor model optimal corresponding to this criterion. In order to demonstrate the power of this method we ran two simulation experiments. In these experiments the optimal rank as well as the correct loadings matrix was found, respectively. Overall, the results are very promising. Further research will include systematic simulations as well as a generalization to nonlinear models.

Acknowledgement

This work has been supported by the Colloborative Research Centre Reduction of Complexity in Multivariate Data Structures (SFB 475) of the German Research Foundation (DFG). We thank Karsten Lübke for technical support.

References

- Bohachevsky, I.O., Johnson, M.E., and Stein, M.L. (1986): Generalized Simulated Annealing for Function Optimization, Technometrics 28, 209-217

- Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. (1992a): Numerical Recipes in C, 2nd ed., Cambridge University Press, Cambridge, 444-455

- Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. (1992b): Numerical Recipes in C, 2nd ed., Cambridge University Press, Cambridge, 408-412

- Reinsel, G., and Velu, R.P. (1998): Multivariate Reduced-Rank Regression, Theory and Applications; Springer, New York

- Röhl, M.C., Weihs, C., and Theis, W. (2002) Direct Minimization of Error Rates in Multivariate Classification; Computational Statistics 17, 29-46;

- Schmidli, H. (1995): Reduced Rank Regression; Physica Verlag