On the convergence of rank-one multi-target linear regression

Pierre Courrieu

On the convergence of rank-one multi-target linear regression

Pierre Courrieu

Centre National de la Recherche Scientifique & Aix-Marseille University, France

Manuscript accepted for publication in ***Statistics*** on February 11, 2021. Cite as:

Courrieu, P. (2021). On the convergence of rank-one multi-target linear regression. *Statistics*, doi: 10.1080/02331888.2021.1891236.

Will be available at: https://doi.org/10.1080/02331888.2021.1891236

Running Head: Multi-target regression

Corresponding author:

Pierre Courrieu

LPC, CNRS-UMR 7290,

CNRS & Aix-Marseille University

3, place Victor Hugo - Bat. 9, Case D

13331 Marseille Cedex 03 – France

Author E-mail: courrieu@free.fr, pierre.courrieu@univ-amu.fr

Abstract. This paper presents a new method for solving rank-one multivariate regression problems, providing a solution that maximizes the sum of squared correlations of the one-dimensional fitted pattern with the target variates. The suitability of the method and the consistency of the estimator are formally proved and experimentally tested. In particular, it is shown that the estimate converges not only as a function of the number of items, but also as a function of the number of target variates. An equivalent conventional reduced-rank regression case is identified, and it inherits the convergence properties of the new approach. Application programs in Matlab/Octave code are provided and numerical examples using artificial data as well as real data are presented.

Key words. multi-target linear regression; multi-output linear regression; reduced rank multivariate regression; regression coefficient estimator consistency

1. Introduction

In this paper, we consider the following maximization problem. Given a random sample of m items, a set of n explicative variables (typically item attributes), and a set of d target variates (typically observation data), let $\Xi \in R^{m \times n}$ be the matrix of regressors, and let $\Theta \in R^{m \times d}$ be the target matrix. Consider the matrix $X \in R^{m \times n}$ whose columns are those of Ξ centred on their mean, and the matrix $T \in R^{m \times d}$ whose columns are those of Θ centred on their mean and normalized to 1. Then find $w \in R^n$ such that:

$$w = \arg\max_{v \in R^n} \Sigma_{k=1..d}\ r^2(Xv, T_k), \qquad (1)$$

where T_k is the kth column of T, and $r(x, y)$ is the Pearson correlation coefficient between vectors x and y. Note that one can as well use Ξ and Θ in (1) instead of X and T, without changing the result since squared correlations are invariant to linear transforms of the variables. This is a multiple linear regression problem if $n>1$, and it is a multi-target linear regression problem if $d>1$. Note that in the multi-target case, only one linear combination of regressors (Xw) is fitted to all the target variates, which is called a "rank-one multi-target linear regression", and in the present case is also "one-dimensional" since the solution is a simple vector. This is different from the common practice where one fits regressors to each target variate independently of other ones. Thus, the question here is not simply to know in what measure the considered regressors can account for each target variate, but rather to know what the target variates have in common that the regressors can account for. This type of approach is known as "multi-target linear regression", "multi-output linear regression", or "multivariate linear regression", and the "rank-one" qualifier refers to the fact that this corresponds to a particular case of the more general "reduced-rank regression"

approach [1, 19,20], where one estimates matrices of regression coefficients of various ranks.

The rank-one, one-dimensional case is of special interest in a number of situations. For instance, one can need to estimate a unique vector of regression coefficients allowing combining a given set of predictive variables in order to predict new target variates and possibly using new items. Such a generalization process is common in machine-learning, however, it can also be useful to build reusable regressors in the context of data analysis. Another important application field in neurosciences is the regression analysis of electrophysiological data such as electroencephalograms (EEG), in particular event related potentials (ERP), where one uses stimulus and/or participant characteristics to explain a sequence of electrophysiological responses at various cortical (scalp) locations [17, 27]. Using a classical multiple regression technique, one can observe significant fits at various locations and time delays in the EEG, while the electrophysiological response patterns are actually very different at these various locations and latencies. Using a rank-one, one-dimensional multi-target regression avoids such a situation by fitting only what is common in the various target response patterns, possibly with different strengths and signs.

Non-linear multi-target regression methods are common in machine-learning studies [3], while linear multi-target regression methods have been more particularly investigated in the framework of multivariate data analysis [1, 4, 5, 19, 20, 26, 29]. As mentioned above, the problem (1) closely relates to the rank-one case of the well-known reduced-rank regression (commonly abbreviated “RRR”) approach [19, 20]. Several variants of the reduced-rank regression have been proposed [26], and recent studies often focus on the determination of an optimal rank for the matrix of regression coefficients [6, 14, 21].

In the present study, however, we a priori fix this rank to one, and we concentrate on certain convergence properties that are relevant for the targeted applications, in particular the convergence of the regression coefficients estimate as a function of the number of target variates. The convergence as a function of the number of items is known [26, section 2.5], however the convergence as a function of the number of targets, independently of the number of items, has not been studied (to our knowledge), and it is commonly assumed that $n+d \leq m$, see [26, p. 3]. We will not retain this assumption, letting d grow freely, and we will define a particular rank-one multi-target regression method that is quite simple and allows us to easily derive the desired convergence properties. We also provide indications for the implementation of this tool, and a ready to use Matlab/Octave code program for applications. Finally, the method and the convergence properties are tested on numerical examples.

T' denotes the transpose of T, and $X^{(1,2,3)}$ denotes any {1,2,3}-inverse of X, such as the unique {1,2,3,4}-inverse of Moore-Penrose [2], or possibly some fast computation {1,2,3}-inverse (Theorem 5 from [9]), for instance. It can be useful to remember the four Penrose equations for the generalized inverse matrices. Let $M^{(k)}$ denote a {k}-inverse of the matrix M, then:

$$MM^{(1)}M = M,\ \ M^{(2)}MM^{(2)} = M^{(2)},\ (MM^{(3)})' = MM^{(3)},\ (M^{(4)}M)' = M^{(4)}M.$$

We will also use the Kronecker (or tensor) product, denoted $A \otimes B$, of matrices A and B, and the column-wise vectorization, denoted vec(A), of a matrix. The notation $[x_i]_{i=1..n}$ corresponds to a column vector of n components (the x_i's).

2. Problem solution

2.1. One-dimensional solution

Theorem 1. Problem (1) is equivalent to the maximal eigen-element problem:

$$XX^{(1,2,3)}TT'y = \lambda_{\max} y, \text{ with } \|y\| = 1,$$

and the solution of problem (1) is:

$$w = X^{(1,2,3)}y, \quad \text{with} \quad \Sigma_{k=1..d}\ r^2(Xw, T_k) = \lambda_{\max}.$$

Proof. Given a vector $v \in R^n$, the kth component of the vector $T'Xv$ is equal to:

$$< Xv, T_k > = \| Xv \| r(Xv, T_k),$$

thus

$$v'X'TT'Xv = \| Xv \|^2 \Sigma_{k=1..d}\ r^2(Xv, T_k). \qquad (2)$$

As a consequence, problem (1) is equivalent to:

$$w = \arg\max_{v \in R^n} v'X'TT'Xv, \text{ subject to } v'X'Xv = 1, \qquad (3)$$

which, at first glance, is a quadratically constrained quadratic programming problem. Setting $z = Xv$, one can reformulate problem (3) as:

$$y = \arg\max_{z \in R^m} z'TT'z, \text{ subject to } \|z\| = 1, \text{ and } XX^{(1,2,3)}z = z, \qquad (4)$$

where the last constraint expresses that z must belong to the range of X, and thus there exists v such that $Xv = z$, that is $v = X^{(1,2,3)}z$. The matrix $XX^{(1,2,3)}$ is an orthogonal projector, it is idempotent (equal to its square) and symmetric [2].

We can now solve problem (4) using the solution of the maximal eigen-element problem of Theorem 1:

with $\|y\| = 1$,

$XX^{(1,2,3)}TT'y = \lambda_{\max} y$

$$\Rightarrow \lambda_{\max} XX^{(1,2,3)}y = XX^{(1,2,3)}XX^{(1,2,3)}TT'y = XX^{(1,2,3)}TT'y$$

$$\Rightarrow y = XX^{(1,2,3)}y \qquad (5)$$

$$\Rightarrow y'TT'y = y'XX^{(1,2,3)}TT'y = \lambda_{\max} y'y = \lambda_{\max} \qquad (6)$$

On the other hand, using (5):

$$y = XX^{(1,2,3)}y \Rightarrow w = X^{(1,2,3)}y \Rightarrow y = Xw$$

$$\Rightarrow w'X'TT'Xw = \Sigma_{k=1..d}\ r^2(Xw, T_k) = \lambda_{\max}\,, \qquad \text{(using (2) and (6))}$$

which completes the proof. □

2.2. Practical considerations

In practice, there are various ways of solving the maximal eigen-element problem of Theorem 1. However, given that we need only the dominant eigenvalue and the corresponding eigenvector, while the problem $m \times m$ matrix, say $A = XX^{(1,2,3)}TT'$, can be huge if the number m of items is very large, it is preferable to use a suitable method based on Krylov subspaces and Arnoldi iterations [23, 30]. However, even in this case, the computation of the $m \times m$ matrix A can be too heavy. Fortunately, the computational complexity can be lowered exploiting the fact that the number n of regressors is usually not very large.

First, we note that in several {1,2,3}-inverse matrix computation methods, $X^{(1,2,3)}$ can be written as $X^{(1,2,3)}=HX'$, where H is a symmetric $n \times n$ real matrix. For instance one can find an H matrix for computing the Moore-Penrose inverse of X in [8], while Theorem 5 from [9] provides an H matrix for computing another type of {1,2,3}-inverse. The first thing to do is to compute the matrix H for the chosen type of {1,2,3}-inverse of X, which is usually a reasonable cost operation. Eigenvalue methods based on Krylov

subspaces use only matrix-vector products, avoiding matrix-matrix products that are very expensive in the case of huge matrices. This principle can be extended to the computation of the problem matrix itself, given that this matrix is used only in matrix-vector products. Let z denote the vector in a matrix-vector product to be computed:

$$Az = XX^{(1,2,3)}TT'z = X(H(X'(T(T'z)))). \quad (7)$$

This way, the computation includes only matrix-vector products, and the possibly huge matrix A is in fact never computed.

One can also note that the sign of the solution vector w depends on the sign of the eigenvector y, and thus it is arbitrary. This determines arbitrarily the signs of the correlation coefficients of Xw with the columns of T, thus the sign of w can possibly be modified in order to satisfy various criteria for the application. For instance, one can choose the sign of the largest correlation, or the sign of the subset of correlations having the same sign and the greatest sum of squares.

2.3 Associated least-squares solution

Well-known methods such as the reduced-rank regressions do not use a correlation criterion to be maximized as in (1), but various weighted least-squares criteria to be minimized [19, 20]. Contrarily to the correlation, the quadratic error is not invariant to linear transformations of the variates, thus in order to minimize it in the rank-one case, the one-dimensional solution of Theorem 1 is not sufficient, and it must be weighted with a specific coefficient while a specific bias coefficient must be append for each target variate. Fortunately, given the one-dimensional solution w of Theorem 1, the required coefficients are very easy to compute. Let $\mu_{\Xi} \in \mathrm{R}^n$ denote the row vector of Ξ column means, let $\mu_{\Theta} \in \mathrm{R}^d$ denote the row vector of Θ column means, and let Y be the

matrix Θ whose columns have been centred (but not normalized), then we must first solve:

$$b' = \arg\min_{v \in R^d} \| Xwv' - Y \|^2 ,$$

where $\| . \|$ denotes the Frobenius norm. The solution is simply the vector:

$$b' = (Xw)'Y / \| Xw \|^2 .$$

The row vector $\mu \in R^d$ of bias coefficients is given by:

$$\mu = \mu_\Theta - \mu_\Xi (wb').$$

Finally, the matrix of least-squares regression coefficient $B \in R^{(n+1) \times d}$ is given by:

$$B = [\mu', (wb')']',$$

where [. , .] here denotes the concatenation operator. The approximation of Θ is:

$$\Theta^* = [\mathbf{1}_m , \Xi] B,$$

and the mean square error (MSE) that will be used in certain numerical test is:

$$\text{MSE} = \| \Theta^* - \Theta \|^2 / (md).$$

Note that the coefficients in b not only change the norm of the regression coefficient vectors, but they can also change their sign independently for each variate. As a result, if one computes the correlations between the columns of Θ^* and those of Θ, all these correlations have the same positive sign, while possible opposite behaviors of different target variates are hidden. This is harmful for applications using correlation statistics, thus in this case the correlations must be computed between the one-dimensional approximation Xw and the columns of Θ (or equivalently of T), which preserves the magnitude of the correlations as well as the correlation sign variations.

Note also that a similar problem occurs in usual reduced-rank regression methods, where the regression coefficients are always in matrix form, even in the rank-

one case. This requires applying a sign correction procedure when one needs to use correlation statistics.

3. Relation with the usual R^2 statistic

A squared Pearson correlation coefficient whose one of the arguments is a linear combination of regressors, as in (1), is similar to a squared multiple correlation coefficient, and it is equal to the well-known R^2 statistic if, and only if, the linear combination of regressors is optimal in the least-squares sense:

$$u_k = \arg\min_{v \in R^n} \| Xv - T_k \|^2 , \quad 1 \le k \le d , \tag{8}$$

which has solutions of the form $u_k = X^{(1,2,3)} T_k$. In this case, the set of optimal regression coefficients is an $n \times d$ matrix $U = X^{(1,2,3)} T$, with one column of regression coefficients per target variate. This provides the maximum possible value to each $R^2(X, T_k)$, $k = 1..d$.

However, in the case of a multi-target regression (1), there is only one vector w of regression coefficients, and we must clarify the relation of w with the columns of U. With w, λ_{max} and y as in Theorem 1, we have the following result:

Lemma 1. Set $U = X^{(1,2,3)} T$, and $c = (\lambda_{max})^{-1} T'y$, then:

If $\lambda_{max} > 0$, then $w = Uc$,

Proof. Using the maximal eigen-element of Theorem 1, we have:

$$XX^{(1,2,3)} TT'y = \lambda_{max}\, y \;\Rightarrow\; XX^{(1,2,3)} Tc = y$$

$$\Rightarrow\; X^{(1,2,3)}\, XX^{(1,2,3)} Tc = X^{(1,2,3)} y = w.$$

On the other hand, since $X^{(1,2,3)}$ is a {2}-inverse, we have:

$$X^{(1,2,3)}\, XX^{(1,2,3)} Tc = X^{(1,2,3)} Tc = Uc,$$

and thus $Uc = w$, which proves Lemma 1. □

Corollary 1. $R^2(X, T_k) = r^2(Xu_k, T_k) \geq r^2(Xw, T_k),\ k = 1..d$.

Proof. The inequality in Corollary 1 results from the fact that the linear combination Uc of the columns of U in Lemma 1 is not necessarily optimal in the least squares sense for the target variates considered individually. □

A consequence of this is that the distribution of the squared correlations in the multi-target case (d>1) does not reduce to the distribution of the well-known R^2 statistic, and thus the F-test associated with the R^2 statistic [7] is not suitable to test the null hypothesis of the r^2 statistics in this case.

Lemma 2. $c_k = r(Xw, T_k) \,/\, (\Sigma_{j=1..d}\ r^2(Xw, T_j)),\ k = 1..d.$

Proof. This is straightforward:

$$c = (\lambda_{max})^{-1}\, T'y \;=\; (\Sigma_{j=1..d}\ r^2(Xw, T_j))^{-1} T'Xw$$

$$= [r(Xw, T_k)]_{k = 1..d} \,/\, (\Sigma_{j=1..d}\ r^2(Xw, T_j)).\ \square$$

4. Statistical significance of the multi-target regression correlation coefficients

4.1 Computing p-values

As we have seen in Section 3 (Corollary 1), one cannot use significance tests associated with usual multiple correlation coefficients to test the significance of multi-

target correlation coefficients. Given that usual parametric tests are not suitable, we will turn to robust distribution-free methods such as permutation tests.

One knows that, in order to build a valid permutation test, it is necessary and sufficient to have exchangeable observations under the null hypothesis [15, 16, 22]. It is usually the case of the items in regression problems, and in particular in problem (1). As we shall see in section 5.1, the used statistical model allows the kth target variate to be written as:

$$T_k = \alpha_k X\omega + \xi_k , \qquad k=1..d$$

where $\alpha_k X\omega$ is the regression part ($\alpha_k \in \mathrm{R}$, $\omega \in \mathrm{R}^n$), and $\xi_k \in \mathrm{R}^m$ is the residual whose components are assumed to be independent and identically distributed (with mean 0 and variance σ^2), which implies that these residue components are exchangeable. On the other hand, the null hypothesis for the kth target variate is:

$$\mathrm{H0}: \alpha_k = 0.$$

Thus, under the null hypothesis, the linear relation of the target with the regressors vanishes, and it remains only the exchangeable residue components (one per item).

Exploiting this, one can use the data to perform repeated Monte-Carlo simulations to generate a large sample from the distribution of multi-target r^2 statistics under the null hypothesis. As we have seen, if the null hypothesis is true, all possible pairings of the regressors values with the target variates values are equally likely to occur. Thus it suffices to repeatedly randomly permute the rows (items) of the matrix X of regressors, or equivalently (but not simultaneously), to repeatedly randomly permute the rows of the target matrix T, then to compute and store the resulting multi-target r^2 statistics. Note, however, that the r^2 distributions associated to the different target variates are not necessarily the same, and we must consider a specific distribution for

each target variate. The estimated p-value for the kth r^2 statistic is the proportion of r^2 statistics in the kth permutation test distribution that are greater or equal to the kth r^2 statistic obtained without permutation.

In order to solve problem (1) for each permutation, let P be a random permutation matrix of order m, then (7) becomes:

$$A_pz = PXX^{(1,2,3)}P'TT'z = X_p(H((X_p)'(T(T'z)))), \quad \text{with } X_p = PX \qquad (9a)$$

or alternatively:

$$A_pz = XX^{(1,2,3)}PTT'P'z = X(H(X'(T_p((T_p)'z)))), \quad \text{with } T_p = PT \qquad (9b)$$

Note, however, that in practice one simply permutes the rows of X or of T using an index permutation operator (what is denoted X_p or T_p), not products with permutation matrices that would be computationally heavy. This procedure (with the option (9b)) is implemented in Matlab/Octave code as the function "MTRegPV" listed in the Appendix. In addition, the listed function "MTRegLS" computes the associated least-squares solution if necessary.

4.2 Controlling for the Family Wise Error Rate

When the number d of target variates is large, there is a potential problem of test inflation that requires a control procedure. A priori, the target variates are related in some way, and the dependences of tests can be of any type. So we must use a method not requiring restrictive hypotheses on the dependences of the tests. This is the case of methods derived from Bonferroni's correction, in particular the Family Wise Error Rate (FWER) control of Holm-Bonferroni [18]. The FWER can be controlled using, for instance, the p-values computed by permutation tests as described in Section 4.1. The FWER control procedure is implemented in Matlab/Octave code as the function "Holm"

listed in the Appendix, and it is optionally called by the function “MTRegPV” if the input argument “fwerc” of “MTRegPV” is set to a strictly positive value.

5. Statistical model and consistency of the estimator

5.1 Statistical model

As usually described in mathematical statistics handbooks (e.g. [25] pp. 237-257), the multiple linear regression model could be summarized as:

$$T_k = X\upsilon_k + \varepsilon_k , \qquad k=1..d \tag{10}$$

where $\upsilon_k \in \mathrm{R}^n$ is the true vector of regression coefficients for the kth target variate, and $\varepsilon_k \in \mathrm{R}^m$ is a vector of residues.

One usually retains the following hypotheses:

H1: $\mathrm{E}(\varepsilon_k) = \mathbf{0}$,

H2: $\mathrm{E}(\varepsilon_k \varepsilon_k') = \sigma^2 \mathrm{I}_m$,

where E is the expected value operator, and I_m the identity matrix of order m.

Proving the consistency of the estimators also requires the additional hypothesis:

H3: $\mathrm{rank}(X) = n$,

which supposes that $m>n$, and the n regressors are linearly independent.

In what concerns the multi-target linear regression model, it is exactly the same as above, except that the regression coefficients must be decomposed as:

$$\upsilon_k = \alpha_k\omega + \delta_k, \quad k=1..d \tag{11}$$

where $\omega \in \mathrm{R}^n$ is the true vector of regression coefficients common to the d target variates, $\alpha_k = \omega' \upsilon_k/(\omega'\omega)$ is the projection coefficient of υ_k on ω, and $\delta_k \in \mathrm{R}^n$ is the residual rejection vector of the kth target variate.

Substituting (11) in (10), one obtains:

$$T_k = X(\alpha_k\omega + \delta_k) + \varepsilon_k = \alpha_k X\omega + (X\delta_k + \varepsilon_k) = \alpha_k X\omega + \xi_k \text{ , } k=1..d \qquad (12)$$

where the quantity $X\delta_k$ is transferred from the regression to the residue.

Concerning the multi-target case, while H3 remains unchanged, we must modify H1 and H2 in the following way. Define the global residue $\xi \in \mathrm{R}^{md}$ as:

$$\xi = \mathrm{vec}(\xi_1, \xi_2, \ldots, \xi_d) \text{ .}$$

This is the residue associated to the target vector $t = \mathrm{vec}(T)$. Modify the hypotheses as:

H1: $\mathrm{E}(\xi) = \mathbf{0}$,

H2: $\mathrm{E}(\xi\,\xi') = \sigma^2 \mathrm{I}_{md}$.

The new versions of the hypotheses imply the old ones. In addition, the new version of H2 supposes that the target variates are linearly independent at the population level.

Note that ω is not defined if the distribution of α_k's concentrates on zero ($\delta(0)$ Dirac distribution). In this case, all multi-target correlations are random, but not necessarily zero (except if all target variates are orthogonal to all regressors). In such a case, we say that the regression problem (1) is "insubstantial".

Definition. We say that the multi-target regression problem (1) is "not insubstantial" if there exists a population parameter $\omega \in \mathrm{R}^{\mathrm{n}}$ and a sample of d coefficients $\{\alpha_k ; k=1..d\}$, as defined in (11), belonging to a population whose probability distribution does not reduce to the Dirac distribution on zero $\delta(0)$. This implies that a non-zero proportion of the coefficient population has a non-zero value.

Intuitively, this means that the considered regressors are not completely irrelevant for the considered population of target variates.

5.2 Consistency of the estimator

Theorem 2. Under the hypotheses H1-H3, consider $w \in \mathrm{R}^n$ as defined in Theorem 1, α_k and ω as defined in (11), and assume that the problem (1) is not insubstantial. Then:

(i) $\mathrm{E}(w) = \omega$,

(ii) $\lim_{d\to\infty} \mathrm{E}(\| w - \omega \|^2) = 0$.

If in addition the regressor variates have finite moments of order 1 and 2, then:

(iii) $\lim_{m\to\infty} \mathrm{E}(\| w - \omega \|^2) = 0$.

Proof. After H3, X is of full column rank, thus all {1,2,3}-inverses of X are equivalent, and one can simply consider $X^\dagger = (X'X)^{-1} X'$.

Let $c \in \mathrm{R}^d$ be defined as in Lemma 1, $t = \mathrm{vec}(T) \in \mathrm{R}^{dm}$, and define $M \in \mathrm{R}^{dm\times n}$ as:

$$M = (c')^\dagger \otimes X.$$

Then consider the following linear least squares system:

$$Mv \approx t \;\Leftrightarrow\; v = M^\dagger t = (c' \otimes X^\dagger)\,\mathrm{vec}(T) = X^\dagger T c = w.$$

Thus, if the vector c is given, then we can transform (1) into an equivalent linear least squares system of matrix M and single target t.

H1 implies that $\mathrm{E}(t) = M\omega$, and one has:

$$\mathrm{E}(w) = (M'M)^{-1} M'\mathrm{E}(t) = (M'M)^{-1} M'M\omega = \omega,$$

which proves (i).

Moreover, let $\Sigma(w)$ denote the covariance matrix of w, then, given H2:

$$\Sigma(w) = (M'M)^{-1} M'(\sigma^2 \mathrm{I})[\,(M'M)^{-1} M']' = \sigma^2\,(M'M)^{-1} M'M(M'M)^{-1} = \sigma^2\,(M'M)^{-1},$$

for some variance σ^2 of the residues.

Now, since $E(w) = \omega$, we have:

$$E(\| w - \omega \|^2) = \text{Trace}(\Sigma(w)) = \sigma^2 \text{ Trace}((M'M)^{-1}).$$

Using Lemma 2, one easily verify that $(c')^\dagger = [r(Xw, T_k)]_{k = 1..d}$, and thus:

$$(M'M)^{-1} = (((c')^\dagger \otimes X)'((c')^\dagger \otimes X))^{-1} = ((\Sigma_{j=1..d}\ r^2(Xw, T_j))X'X)^{-1}$$

$$= (\Sigma_{j=1..d}\ r^2(Xw, T_j))^{-1}\ (X'X)^{-1}.$$

This implies that:

$$E(\| w - \omega \|^2) = \sigma^2 \text{ Trace}((X'X)^{-1})/\ (\Sigma_{j=1..d}\ r^2(Xw, T_j)), \qquad (13)$$

where the numerator does not depend on d, and the denominator ($= \lambda_{\max}$) tends to infinity as d tends to infinity, because the problem (1) is not insubstantial, by hypothesis, thus the number of non-zero coefficients α_k's tends to increase proportionally to d and the resulting sum of squared correlations tends to infinity. This proves (ii).

Given that the regressors are centred, $X'X/m$ is a convergent estimator of the true covariance matrix V of the regressor variates, which is regular after H3. Thus:

$$\lim_{m\to\infty} X'X/m = V \Rightarrow$$

$$\lim_{m\to\infty} (X'X/m)^{-1} = V^{-1} \Rightarrow$$

$$\lim_{m\to\infty} m(X'X)^{-1} = V^{-1} \Rightarrow$$

$$\lim_{m\to\infty} \text{Trace}((X'X)^{-1}) = \lim_{m\to\infty} \text{Trace}(V^{-1})/m = 0.$$

Substituting this in (13), one obtains (iii), which completes the proof of Theorem 2. □

Remark. As noted in section 2.2, the sign of the eigenvector y in Theorem 1 is arbitrary, that is, if y is a solution, then $-y$ is also a solution. As a result, the sign of w, the sign of c

(Lemma 1), and the sign of the correlations (Lemma 2) are also arbitrary. Now, considering the unknown vector ω as the asymptotic solution of problem (1), it is clear that the sign of ω is also arbitrary. This indeterminacy is lifted in the above proof because the vector c is used in the calculation of the matrix M, thus the sign of ω is implicitly the sign corresponding to that of c and to that of w.

Note that (13) shows how and why increasing the number of correlated linearly independent target variates (d) enhances the convergence of the estimator (w).

Now, if rank(X) < n , for instance because $m \le n$, then H3 is not met and Theorem 2 does not apply. However, Theorem 1 remains valid and the method still works, but the estimator is generally biased since $\mathrm{E}(w) = M^{(1,2,3)}\mathrm{E}(t) = M^{(1,2,3)}M\omega$, which is usually different from ω when M is not of full column rank.

6. Relation with reduced-rank regression methods

Let $X \in R^{m\times n}$ and $Y \in R^{m\times d}$ be the centred columns versions of the regressor and target matrices, respectively, and let $T \in R^{m\times d}$ be the normalized columns version of Y, as previously.

Definition 1. Set $s_k = a\| Y_k \|$, k=1..d, for any fixed real a>0. Then define $S \in R^{d\times d}$ as the diagonal matrix whose diagonal elements are the s_k 's, k=1..d.

Theorem 3.

The following rank-one reduced-rank regression problem with S^{-2} weighting:

find $v \in R^n$, and $q \in R^d$ that minimize Trace{$(Xvq' - Y)S^{-2}(Xvq' - Y)'$}

can be solved using Theorem 1 since it is equivalent to :

(i) find $v \in R^n$ that maximizes $v'X'TT'Xv$, subject to $v'X'Xv = 1$,

(ii) then set $q' = (Xv)'Y$.

Conversely, let $Q \in R^{n\times d}$ be the matrix of regression coefficients provided by the rank-one RRR with S^{-2} weighting, and let Q_k be its kth column, $1 \le k \le d$, then:

(iii) $v = \pm Q_k / \| XQ_k \|$.

Proof. First, suppose that one knows v, then the solution of:

$$\arg\min_{q \in R^d} \mathrm{Trace}\{(Xvq' - Y)S^{-2}(Xvq' - Y)'\} = \arg\min_{q \in R^d} \| (Xvq' - Y)S^{-1} \|^2$$

is simply given by:

$$q' = (Xv)'Y / \| Xv \|^2 = (Xv)'Y,$$

which proves (ii).

Next, one must determine v. One has:

$$\min_{v,q} \mathrm{Trace}\{(Xvq' - Y)S^{-2}(Xvq' - Y)'\} = \min_{v,q} \mathrm{Trace}\{ S^{-1}(Xvq' - Y)'(Xvq' - Y) S^{-1}\} =$$

$$\min_{v,q} \mathrm{Trace}\{ S^{-1}qv'X'Xvq'S^{-1} - S^{-1}Y'Xvq'S^{-1} - S^{-1}qv'X'YS^{-1} + S^{-1}Y'YS^{-1} \} =$$

$$\min_{v} \mathrm{Trace}\{ S^{-1}\| Xv \|^{-2} Y'Xv \| Xv \|^2 v'X'Y \| Xv \|^{-2}S^{-1}$$

$$- S^{-1}Y'Xvv'X'Y \| Xv \|^{-2}S^{-1} - S^{-1}\| Xv \|^{-2}Y'Xvv'X'YS^{-1} + S^{-1}Y'YS^{-1}\} =$$

$$\min_{v} \mathrm{Trace}\{S^{-1}\| Xv \|^{-2}Y'Xvv'X'Y S^{-1} - 2 S^{-1}\| Xv \|^{-2}Y'Xvv'X'YS^{-1} + S^{-1}Y'YS^{-1}\} =$$

$$\min_{v} \mathrm{Trace}\{ S^{-1}Y'YS^{-1} - \| Xv \|^{-2}S^{-1}Y'Xvv'X'YS^{-1}\} =$$

$$\min_{v} \mathrm{Trace}\{ a^{-2}T'T - a^{-2}\| Xv \|^{-2}T'Xvv'X'T\} =$$

$$\min_{v} \mathrm{Trace}\{ a^{-2}TT' - a^{-2}\| Xv \|^{-2}v'X'TT'Xv\},$$

whose solution is given by:

$$\arg\max_{v \in R^n} (v'X'TT'Xv), \text{ subject to } \| Xv \|^2 = 1,$$

which is similar to (i) and to equation (3), and thus can be solved using Theorem 1. Finally, if $Q = vq'$, then all columns of Q are collinear to v, while the constraint $\| Xv \| = 1$ implies (iii). Note, however, that the solution sign is always arbitrary. □

Corollary 2. The consistency properties stated in Theorem 2 for the method of Theorem 1 also apply to the rank-one reduced-rank regression with a weight matrix of type S^{-2}.

Proof. This immediately follows from Theorem 3 since the two considered methods are equivalent. □

7. Computational tests

7.1 Generating artificial data

Using artificial data allows us to know the exact solution of problems and thus to compare the results provided by various methods with the exact ones. In the present case, we use the following model to generate each column of the target matrix.

$$T_k = \alpha_k X\omega + \beta_k b + \gamma_k g_k, \quad k=1..d \qquad (14)$$

where $X \in \mathrm{R}^{m\times n}$ is the matrix of regressors, $\omega \in \mathrm{R}^n$ is the vector of exact regression coefficients, $b \in \mathrm{R}^m$ with $b'X=\mathbf{0}$ is a shared factor not accountable for by the regression, and $g_k \in \mathrm{R}^m$ is a random Gaussian component of mean $\mathbf{0}$. The column vectors $X\omega$, b, and g_k have unit norms and the real coefficients α_k, β_k, and γ_k are such that $\alpha_k^2 + \beta_k^2 + \gamma_k^2 = 1$. Varying α_k allows one to approximately control the kth exact correlation value $\mathrm{r}(X\omega, T_k)$, but this one must be precisely computed afterwards.

7.2 Typical observations

We used (14) to generate data with m=300, n=30, d=30, the α_k's were uniformly sampled in various intervals, and the β_k's were set to zero, the γ_k's providing the complements to the α_k's. We studied the multi-target regression r-values, computed with the "MTRegPV" function listed in the Appendix, and the classical multiple regression R values, as functions of the exact r values. The statistical significance was established at the risk $\alpha = 0.05$, using classical parametric F-tests for the exact r's and the multiple regression R's, while we used permutation tests as described in section 4.1 for the multi-target regression r-values. The FWER control was not used for the basic studies, but it was controlled afterwards.

Figure 1 shows a typical result with the α_k's sampled in the interval [-0.4, 0.7]. As one can see, the multi-target r-values (r_{MT}) were very close to the exact r values (r_{exact}), with a regression line of equation $r_{MT} \approx 1.0077\ r_{exact} - 0.0007$, (fit r = 0.9999). The correlation of the multi-target regression coefficients w with the exact coefficients ω was r=0.9925. In what concerns the classical multiple regression R, one can see that it substantially overestimated the exact positive correlations, as well as the absolute value of negative ones (remember that the R coefficient is not signed). There were 25 significant exact correlation coefficients, 22 significant multi-target correlation coefficients, and 18 significant multiple regression R coefficients, without FWER control. After FWER control, it remained respectively 22, 16, and 16 significant coefficients.

The results presented in Figure 2 were obtained with the α_k's sampled in the interval [0, 0], that is, all coefficients were random. In this case, significant coefficients are type I errors. Without FWER control, 2 exact correlation coefficients were significant (6.7%) at the 0.05 risk, only 1 multi-target correlation coefficient was significant (3.3%),

and 3 multiple regression R's were significant (10%). With FWER control, there was no longer significant coefficient.

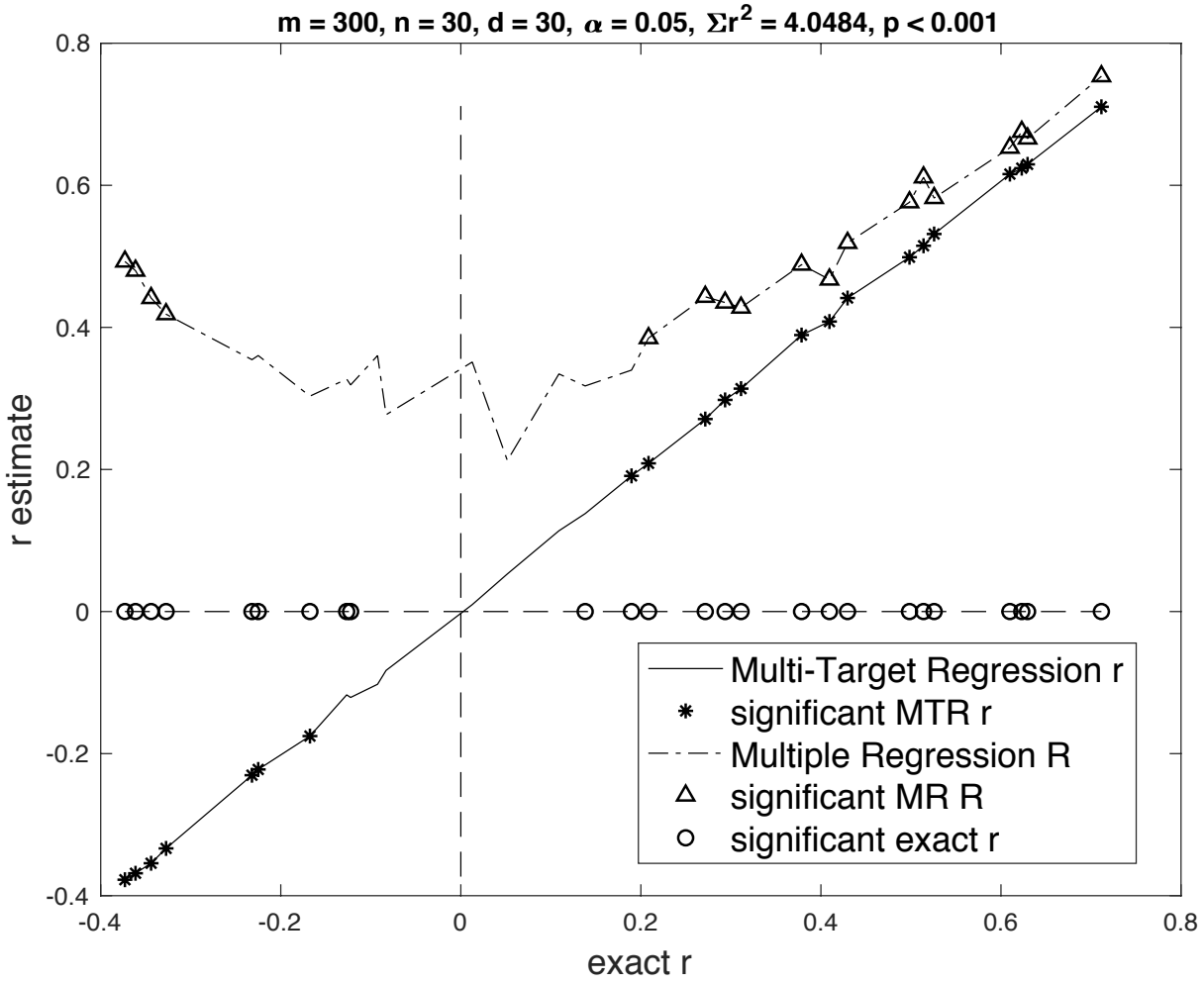

Figure 1. Multi-target linear regression r coefficients and classical multiple regression R coefficients as functions of the exact r coefficients. The data were generated using the model (14) with $\alpha_k \in [-0.4, 0.7]$, $\beta_k = 0$, k=1..30. The α–risk was set to 0.05 and there was no FWER control.

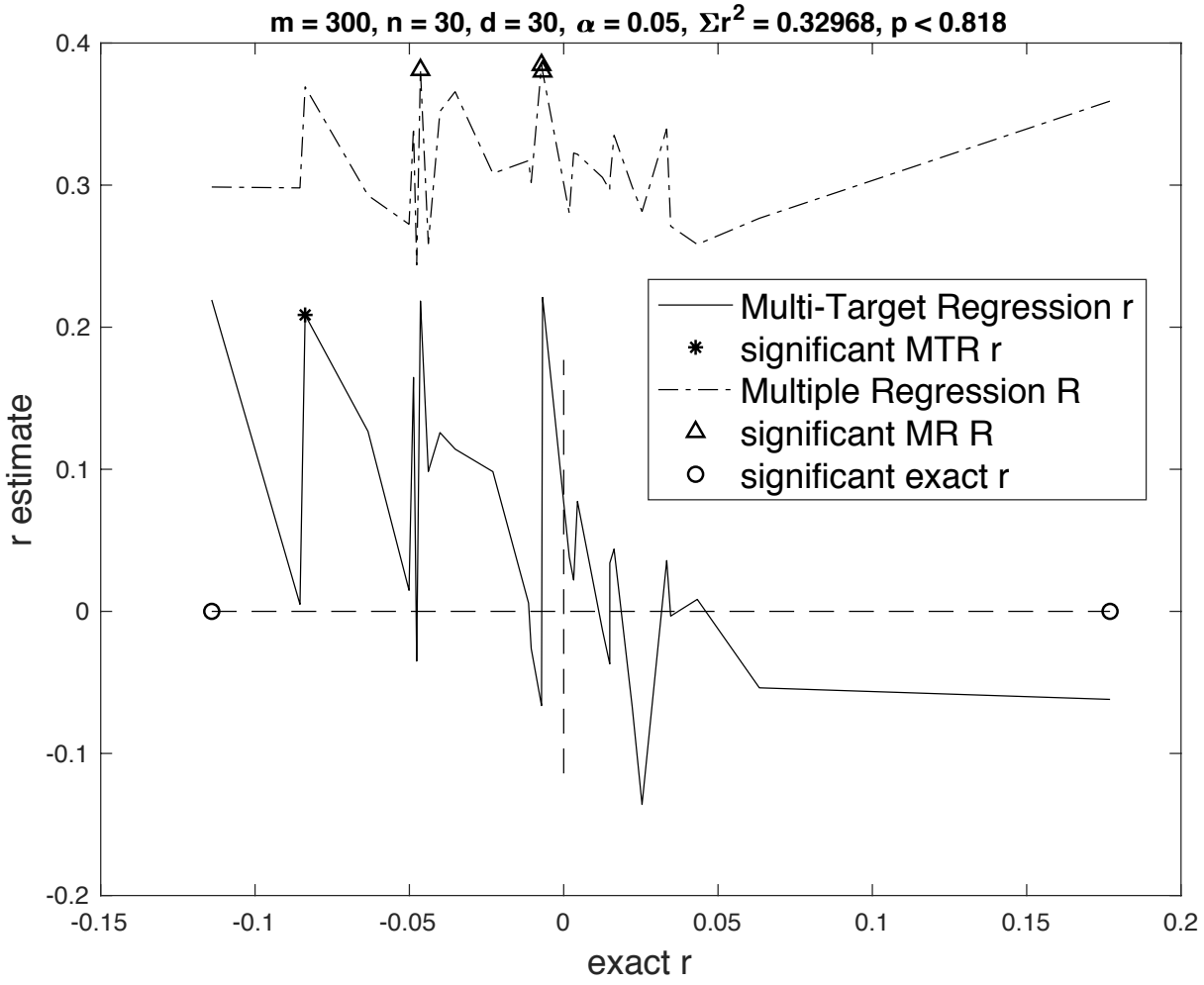

Figure 2. Multi-target linear regression r coefficients and classical multiple regression R coefficients as functions of the exact r coefficients. The data were generated using the model (14) with $\alpha_k = 0$, $\beta_k = 0$, k=1..30. The α–risk was set to 0.05 and there was no FWER control.

7.3 Effect of the number of target variates

In order to see how the number of target variates and the number of items influence the regression accuracy in the framework of the data-generating model (14), we varied the parameter d from 8 to 128 (in powers of 2), the parameter $m \in \{160, 320\}$, while the other parameters were set to n=24, $\alpha_k \in$ [-0.4, 0.7], $\beta_k = 0$, k=1..d, and the α-risk was set to 0.05. For each number of target variates and of items, we generated

and analysed 100 independent random data sets, comparing the obtained regression coefficients (w) with the exact ones (ω), and comparing the multi-target r coefficient values together with their significance to those of the exact r coefficients.

Instead of being arbitrarily chosen, the sign of w was determined in such a way that $w'\omega \geq 0$. Then the accuracy of w was measured by the error norm $\| w - \omega \|$.

The accuracy of the estimated r coefficients was measured using their "discrepancy" (D) from the exact r-values:

$$D(r_{MT}, r_{exact}) = \max_{k=1..d} |r_{MT}(k) - r_{exact}(k)| .$$

The obtained statistical significance (at the .05 level) was also compared to that of the exact coefficients without FWER control. We computed the "significance agreement" defined as the proportion of target variates for which both the estimate and the exact coefficient lead to the same conclusion (significant or non-significant).

Table 1 shows the averages over 100 tests of these accuracy measures as functions of the number of target variates and of the number of items. The average sums of squared correlations (λ_{max}) and the average computation times (in seconds) for the multi-target regressions are also reported (for Matlab 9.4 on Mac OS X 10.11.6). For each performance measure, a one-factor standard analysis of variance was performed to test the effect of the number of target variates (d). The obtained significances are reported in the last column of Table 1. As one can see in Table 1, regardless of m, both the w error and the r_{MT} discrepancy significantly decreased, converging towards 0, while the significance agreement converged towards 1, as the number of target variates increased. In the same time both λ_{max} and the computation time increased. As expected, the accuracy of the estimates was better with 320 items than with 160 items.

Table 1. Variation of the averages (over 100 tests) of the accuracy measures for the multi-target linear regression, as functions of the number of target variates (*d*) and the number of items (*m*). The data were generated using the model (14) with *n*=24, $\alpha_k \in [-0.4, 0.7]$, $\beta_k = 0$, *k*=1..*d*. The α–risk was set to 0.05 and there was no FWER control.

Number of target variates:	*d* = 8	*d* = 16	*d* = 32	*d* = 64	*d* =128	*d* effect significance
m = 160, *n* = 24						
λ_{max}	1.3427	2.3426	4.3936	8.3685	16.542	p < .0001
$\|\| w - \omega \|\|$	0.0248	0.0184	0.0134	0.0101	0.0069	p < .0001
r_{MT} discrepancy from exact r	0.0510	0.0391	0.0298	0.0234	0.0182	p < .0001
Significance agreement	0.7100	0.7712	0.8419	0.8858	0.9083	p < .0001
Computation time (seconds)	1.3125	1.3591	1.7974	2.1251	2.7400	p < .0001
m = 320, *n* = 24						
λ_{max}	1.2844	2.2700	4.2678	8.2755	16.333	p < .0001
$\|\| w - \omega \|\|$	0.0114	0.0089	0.0064	0.0047	0.0034	p < .0001
r_{MT} discrepancy from exact r	0.0213	0.0183	0.0147	0.0110	0.0091	p < .0001
Significance agreement	0.7925	0.8675	0.9187	0.9402	0.9625	p < .0001
Computation time (seconds)	1.7061	1.8999	2.2446	2.9617	3.9808	p < .0001

Table 2 shows the averages over 100 tests of the accuracy measures as functions of the number of target variates and of the number of regressors. Large values, greater than the number of items, have been selected for the number of targets. One of the two numbers of regressors is greater than the number of items, resulting in a violation of the hypothesis H3. As one can see in Table 2, regardless of *n*, both the *w* error and the r_{MT} discrepancy significantly decreased as the number of target variates increased. However, we a priori know that in the case where *n*>*m* the estimator is biased, resulting in a non-zero error limit. We also observe in this case that the significance agreement is quite bad and does not improve as *d* increases.

Table 2. Variation of the averages (over 100 tests) of the accuracy measures for the multi-target linear regression, as functions of the number of target variates (d) and the number of regressors (n). The data were generated using the model (14) with m=64, $\alpha_k \in$ [-0.4, 0.7], β_k = 0, k=1..d. The α–risk was set to 0.05 and there was no FWER control.

Number of target variates:	d=64	d=128	d=256	d=512	d=1024	d effect significance
m = 64, n = 24						
λ_{max}	8.9186	17.535	34.421	68.511	136.40	p < .0001
$\| w - \omega \|$	0.0310	0.0220	0.0158	0.0109	0.0077	p < .0001
r_{MT} discrepancy from exact r	0.0606	0.0473	0.0361	0.0256	0.0191	p < .0001
Significance agreement	0.7142	0.7303	0.7438	0.7495	0.7501	p < .0001
Computation time (seconds)	1.6418	1.8679	2.5013	3.4486	5.8270	p < .0001
m =64, n =96 (H3 violation)						
λ_{max}	9.5619	18.154	34.960	68.885	137.02	p < .0001
$\| w - \omega \|$	0.0918	0.0836	0.0817	0.0796	0.0750	p < .0001
r_{MT} discrepancy from exact r	0.1051	0.0782	0.0592	0.0428	0.0331	p < .0001
Significance agreement	0.4330	0.4264	0.4361	0.4348	0.4332	p < .25, ns
Computation time (seconds)	1.7699	2.0180	2.6552	3.5967	6.0494	p < .0001

Globally, all these observations perfectly illustrate Theorem 2. In summary, one can say that in order to obtain a good rank-one estimate, one must use as many items and target variates as possible, while always having a number of items greater than the number of regressors and using linearly independent regressors. We also know that the target variates must be correlated but linearly independent at the population level. For instance, it is intuitively clear that if one artificially increases the number of targets by repeating many times the same small set of variates, this provides no more information, and there is no reason that this improves the convergence. Note however that, at the data sample level, the target vectors (T columns) cannot be linearly

independent if the number of targets is greater than the number of items, but this does not prevent the convergence, as illustrated in Table 2.

7.4 Comparison with reduced-rank regressions on controlled rank artificial data

In this section, we compare the behaviour of the proposed rank-one regression method with that of conventional reduced-rank regression methods, on rank-one and rank-three artificial data. The data were generated using a random 615×24 matrix of regressors and a random 24×100 coefficient matrix whose rank was controlled (using the singular value decomposition) to be 1 or 3. Target variates were obtained multiplying the regressors by the coefficient matrix and adding Gaussian noise. An available Matlab implementation of Izenman's reduced-rank regression (RRR) was used to test alternative methods [24]. We used the rank-one RRR and the 5-folds cross-validation optimized rank RRR, both with the S^{-2} weight matrix, the identity weight matrix (I), and the $\mathrm{cov}(Y)^{-1}$ weight matrix ("canonical").

Table 3 reports, for each case, the sum of squared correlations (Σr^2), the mean squared error (MSE), the obtained solution rank, and the Bayesian Information Criterion (BIC). The BIC is commonly used as a model selection criterion [26, 28], while the model having the lowest BIC must be preferred.

As one can see in Table 3, the new method (noted "MTR") and the rank-one RRR with S^{-2} weighting have exactly the same performance (as expected from Theorem 3). For rank-one data, these methods provided the maximum sum of squared correlations, while the minimum MSE was obtained with the identity weighting, which also resulted in the minimum BIC. For rank-three data, the optimized rank RRRs detected the appropriate rank, leading these methods to the best performance, including the BIC.

Table 3. Numerical test on artificial data with m=615, n=24, d=100, and the actual rank of the true regression coefficient matrix is 1 or 3. The compared methods are the MTR (Theorem 1), the rank-one RRR or the 5-folds, cross validated rank (CV-rank) RRR, both with the S^{-2} weight matrix, the identity weight matrix (I), and the $\mathrm{cov}(Y)^{-1}$ weight matrix (Canonical). The table reports the sum of squared correlations (Σr^2), the mean squared error (MSE), the obtained solution rank, and the Bayesian Information Criterion (BIC).

	Actual rank = 1				Actual rank = 3			
Method	Σr^2	MSE	rank	BIC	Σr^2	MSE	rank	BIC
MTR	22.7174	6.2106	1	1.8483	28.9348	6.9650	1	1.9629
Rank-1 RRR:								
S^{-2} weight	22.7174	6.2106	1	1.8483	28.9348	6.9650	1	1.9629
I weight	22.7039	6.2095	1	1.8481	28.8590	6.9569	1	1.9618
Canonical	22.7148	6.2110	1	1.8484	28.9297	6.9643	1	1.9629
CV-rank RRR:								
S^{-2} weight	22.7174	6.2106	1	1.8483	42.8318	5.6069	3	1.7891
I weight	22.7039	6.2095	1	1.8481	42.7966	5.6035	3	1.7885
Canonical	22.7148	6.2110	1	1.8484	42.8187	5.6084	3	1.7894

7.5 Comparison with reduced-rank regressions on real data

In this section, we compare the performance of the same methods as in the previous section, but on a real data regression problem whose actual rank is a priori unknown. The data are z-scores of speeded naming latencies of 615 printed French words, provided by 100 human participants. These data were previously studied in [11, 12]. The purpose here is to study the direct effect of the orthographic form of the words on the naming latencies. In order to do this, we need a suitable numerical representation of the character strings that could be used as a multidimensional regressor. A possible numerical coding of character strings has previously been proposed in [10] as the output layer of artificial neural networks for handwriting recognition, and was also used in [27] as a multidimensional orthographic regressor to analyse cerebral event related potentials (EEG/ERP) in a printed word naming task. The coding principle is as follows.

Consider an alphabet of N symbols $\{s_1, s_2, ..., s_N\}$, for instance the 26 lower-case letters of the Roman alphabet. The coding associates to each symbol of the alphabet one component of a real vector $(c_1, c_2, ..., c_N)$. Let χ be a symbol string of L characters, one first determines the "symbol position bits" as $b_{k,i} = 1$ if the symbol s_i appears at rank k in χ, else one has $b_{k,i} = 0$. Then the components of the orthographic code are given by:

$$c_i(\chi) = (\Sigma_{k=1..L}\, b_{k,i}\, 2^{-k})^p \ , \ i=1..N,\ 0 < p \leq 1, \qquad (15)$$

where p is a free parameter (we use p=1/3). Such a numerical code unequivocally represents the corresponding character string and it can always be decoded back into this string.

The necessary alphabet for describing the 615 experimental words had only 24 letters since “k” and “w” are very rare in French, and these two letters never appeared in the corpus. Moreover, the used words did not include letters with diacritic marks. Thus the regressor matrix was a 615×24 real matrix of numerical string codes (15).

One can observe in Table 4 that the rank-one RRR with identity matrix weighting provided the same performance as those of the MTR and the rank-one S^{-2} weighted RRR. This is because the data are z-scores, thus all columns of the target matrix have the same norm and the identity matrix, in this case, is also a S^{-2} matrix. Optimized rank RRRs detected a rank-two case, however, the BIC suggests that the best choice is in fact the MTR model and its equivalent rank-one RRRs. In other words, the improvement of the MSE does not compensate the increase of model complexity with rank-two solutions, and these observations finally suggest that a rank-one model is the most suitable. In fact, one knows that this type of data is essentially described by a general rank-one model, but with an additional small amount of non-random idiosyncratic effect that could explain the detected rank-two solution [13]. A common practice is to calculate the

average response vector over all participants, and then to fit this vector with a linear combination of regressors, using an ordinary multiple regression. Doing this, one implicitly assumes an underlying model where the α_k's in (12) are equal for all participants, which is possibly unrealistic and can lead to problematic results. Using the multi-target regression avoids this drawback and can provide a much better solution.

Table 4. Comparison of 7 multivariate regression methods on the analysis of real z-scores of naming latencies of m=615 printed words, by d=100 human participants, regressed on numerical orthographic codes of the words with n=24 components. The 100 correlation coefficients provided by MTR (Theorem 1) and equivalent methods are in the range [0.1306, 0.5044] and significant at the 0.05 level, with a mean r of 0.3225.

Method	Σr^2	MSE	rank	BIC
MTR	10.9564	0.8890	1	-0.0956
Rank-1 RRR:				
S^{-2} weight	10.9564	0.8890	1	-0.0956
I weight	10.9564	0.8890	1	-0.0956
Canonical	10.1679	0.8969	1	-0.0868
CV-rank RRR:				
S^{-2} weight	11.7183	0.8814	2	-0.0825
I weight	11.7183	0.8814	2	-0.0825
Canonical	11.4674	0.8839	2	-0.0797

7.6 Observing the estimator convergence with real data

With real data, such as those used in the previous section, one does not know the exact solution. However one can exploit the fact that if two independent estimates converge on the same solution, then they also converge on each other. The idea here is that if the used real data fulfil the assumptions of Theorem 2, then one should observe the predicted convergence as a function of d and as a function of m.

Using the regressors and the behavioural data of section 7.5, we repeatedly sampled pairs of independent random samples of d = 10, 20, 30, 40, or 50 participants, we computed the regression solution (Theorem 1) for each of the two samples (w_1 and w_2), and the Euclidean distance between these two solutions. This was repeated 100 times for each modality of d, and a standard analysis of variance was performed on the solution distances. In addition, the same was done not varying d but the number m of items, with m = 61, 122, 183, 244, or 305. The results are reported in Table 5, where one can clearly observe the convergence as a function of d and as a function of m.

Table 5. Average distance of two independent estimates of the regression coefficient vector w (from Theorem 1), as a function of the number d of target variates (participants), and of the number m of items (words). The regressors are the 24 dimensions of a numerical orthographic code (15), and the target data are z-scores of printed word naming times. Each test was repeated 100 times, and a standard analysis of variance was performed for each factor (d and m).

m=615, d:	10	20	30	40	50	d effect
$\| w_1 - w_2 \|$	0.1046	0.0784	0.0637	0.0535	0.0492	p < .0001
d=100, m:	61	122	183	244	305	m effect
$\| w_1 - w_2 \|$	0.7853	0.5401	0.4101	0.3223	0.2605	p < .0001

8. Conclusion

We have defined a new method for solving multivariate linear regression problems, with only one shared vector of regression coefficients. This way, only what is common to the target variates is fitted while the sum of squared correlations of the fitted pattern with the target variates is maximized. This is equivalent to a rank-one

variant of Izenman's reduced rank regression method [19, 20], with a weighting proportional to a diagonal inverse-variance matrix. However, the formulation of the new method allowed us to derive certain new convergence properties of interest for practical applications. In particular, one clearly shows how and why the estimation of the regression coefficients converges as a function of the number of target variates, as well as a function of the number of items. Of course, these consistency properties also concern the equivalent rank-one, reduced-rank regression method, however, we don't know whether or not the convergence as a function of the number of target variates concerns other RRR approaches. A ready to use Matlab/Octave code program is provided for practical applications, and numerical tests on artificial data as well as on real data fully corroborated the formal statements.

Funding: This work, carried out within the Labex BLRI (ANR-11-LABX-0036) and the Institut Convergence ILCB (ANR-16-CONV-0002), has benefited from support from the French government, managed by the French National Agency for Research (ANR) and the Excellence Initiative of Aix-Marseille University (A*MIDEX)

References

[1] Anderson, T.W. (1951). Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Annals of Mathematical Statistics , 22*, 327–351.

[2] Ben-Israel, A., & Greville, T.N.E. (2003). *Generalized Inverses: Theory and Applications, 2nd ed.*, Springer-Verlag, New York.

[3] Borchani, H., Varando, G., Bielza, C., & Larrañaga, P. (2015). A survey on multi-output regression. *WIREs Data Mining Knowledge Discovery, 5(5)*, 216–233. doi: 10.1002/widm.1157

[4] Breiman, L., Friedman, J.H. (1997). Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *59(1)*, 3–54.

[5] Brillinger, D.A. (1981). *Time Series: Data Analysis and Theory (Expanded Ed.)*. Holden Day, Inc., San Francisco. Republication: 2001, SIAM, Philadelphia.

[6] Buena, F., She, Y., & Wegkamp, M.H. (2011). Optimal selection of reduced rank estimators of high-dimensional matrices. *The Annals of Statistics, 39(2)*, 1282–1309. DOI: 10.1214/11-AOS876

[7] Cohen, J., Cohen, P., West, S.G., & Aiken, L.S. (2003). *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences (3rd Ed.)*. London, Lawrence Erlbaum Associates, Publishers.

[8] Courrieu, P. (2005). Fast computation of Moore-Penrose inverse matrices. *Neural Information Processing - Letters and Reviews, 8(2)*, 25-29. (https://arxiv.org/ftp/arxiv/papers/0804/0804.4809.pdf)

[9] Courrieu, P. (2009). Fast solving of Weighted Pairing Least-Squares systems. *Journal of Computational and Applied Mathematics*, *231*, 39-48.

[10] Courrieu, P. (2012). Density Codes, Shape Spaces, and Reading. *ERMITES 2012 : Representations and Decisions in Cognitive Vision*. La Seyne-sur-Mer, August 30-31 and September 1. Proceedings : http://glotin.univ-tln.fr/ERMITES12/

[11] Courrieu, P., Brand-D'Abrescia, M., Peereman, R., Spieler, D., Rey, A. (2011). Validated intraclass correlation statistics to test item performance models. *Behavior Research Methods, 43*, 37-55. doi: 10.3758/s13428-010-0020-5

[12] Courrieu, P., & Rey, A. (2011). Missing data imputation and corrected statistics for large-scale behavioral databases. *Behavior Research Methods, 43*, 310-330. doi: 10.3758/s13428-011-0071-2

[13] Courrieu, P., & Rey, A. (2015). General or idiosyncratic item effects: what is the good target for models? *Journal of Experimental Psychology: Learning, Memory, and Cognition, 41(5)*, 1597-1601. DOI: 10.1037/xlm0000062

[14] Giraud, C. (2011). Low rank multivariate regression. *Electronic Journal of Statistics*, *5*, 775-799. DOI:10.1214/11-EJS625

[15] Good, P. (2002). Extensions of the concept of exchangeability and their applications. *Journal of Modern Applied Statistical Methods, 1(2)*, Article 34. DOI: 10.22237/jmasm/1036110240, http://digitalcommons.wayne.edu/jmasm/vol1/iss2/34

[16] Good, P. (2005). *Permutation, Parametric, and Bootstrap Tests of Hypotheses, 3rd ed.*, Springer Series in Statistics, New York.

[17] Hauk, O., Davis, M.H., Ford, M., Pulvermüller, F., & Marslen-Wilson, W.D. (2006). The time course of visual word recognition as revealed by linear regression analysis of ERP data. *NeuroImage, 30*, 1383–1400.

[18] Holm, S. (1979). A Simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics, 6(2)*, 65-70.

[19] Izenman, A.J. (1975). Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, *5*, 248-264.

[20] Izenman, A.J. (2013). *Modern Multivariate Statistical Techniques*. Springer Texts in Statistics, New York, Springer Science+Business Media. DOI 10.1007/978-0-387-78189-1_6

[21] Kargin, V. (2015). On estimation in the reduced-rank regression with a large number of responses and predictors. *Journal of Multivariate Analysis, 140*, 377-394. https://doi.org/10.1016/j.jmva.2015.06.004

[22] Legendre, P., & Legendre, L. (1998). *Numerical ecology, 2nd English edition*. Elsevier Science BV, Amsterdam.

[23] Lehoucq, R.B., Sorensen, D.C., & Yang, C. (1998). *ARPACK Users Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods.* SIAM, Philadelphia, PA.

[24] McComb, C. (2018). Reduced Rank Regression. *File Exchange – MATLAB Central*. https://mathworks.com/matlabcentral/fileexchange/53024-reduced-rank-regression

[25] Montfort, A. (1982). *Cours de Statistique Mathématique*. Economica, Paris.

[26] Reinsel, G.C., & Velu, R.P. (1998). *Multivariate Reduced-Rank Regression: Theory and Applications.* Springer: Lecture Notes in Statistics, New York.

[27] Rey, A., Madec, S., Grainger, J., & Courrieu, P. (2013). Accounting for variance in single-word ERPs. Oral communication presented at the *54th Annual Meeting of the Psychonomic Society*, Toronto, Canada, November 14-17.

[28] Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, *6*, 461-464.

[29] Similä, T., & Jarkko, T. (2007). Input selection and shrinkage in multiresponse linear regression. *Computational Statistics & Data Analysis*, *52*, 406–422. doi:10.1016/j.csda.2007.01.025

[30] Sorensen, D.C. (2002). Numerical methods for large eigenvalue problems. *Acta Numerica*, 519-584. DOI: 10.1017/S0962492902000089

Appendix. Matlab/Octave code of useful functions (for academic use only)

```
function [sr2,psr2,r,p,s,w] = MTRegPV(X,T,fwerc,a)
% Multi-Target Multiple Linear Regression
% For m items, n regressors, and d target variates
% p-values are computed by permutation test
%             Input arguments:
% X: m x n matrix of regressors
% T: m x d matrix of target variates
% fwerc: control the FWER (1) or not (0)
% a: chosen alpha risk (default: a=0.05)
%             Output arguments:
% sr2: sum of squared correlations
% psr2: p-value of sr2 (by permutation test)
% r: d x 1 vector of r estimates
% p: d x 1 vector of p-values (by permutation test)
% s: d x 1 significance vector (1=signif., 0=n.s.)
% w: n x 1 vector of regression coefficients
% ------------------------------------------------
[m,n]=size(X); [mt,d]=size(T);
if mt ~= m, error('Matrix dimension error'); end
if nargin<4, a=0.05; end
if (nargin<3) || (fwerc<=0)
    np=max(ceil(1/a),1000);
else
    np=max(ceil(d/a),1000);
end
if np>10000
warning(['number of permutations = ',num2str(np)]);
end
for j=1:n
    X(:,j)=X(:,j)-mean(X(:,j));  % centre X columns
end
for k=1:d          % centre and normalize T columns
    T(:,k)=T(:,k)-mean(T(:,k));
    T(:,k)=T(:,k)/norm(T(:,k));
end
H = pinv(X'*X);              % compute the solution
perm = (1:m);
MVfcn = @(x) X*(H*(X'*(T(perm,:)*(T(perm,:)'*x))));
[y,sr2] = eigs(MVfcn,m,1);
w = H*(X'*y);                          % then Xw = y
r = (y'*T)';
v2=[sum(r(r>=0).^2),sum(r(r<0).^2)];  % sign choice
if v2(1)<v2(2)
    w=-w;
    r=-r;
end
```

```
Dr2 = zeros(np,d);       % permutation test of H0's
Dsr2 = zeros(np,1);
Dr2(1,:)=(r.^2)'; Dsr2(1)=sr2;
for t=2:np
    perm = randperm(m);
    MVfcn = @(x) X*(H*(X'*(T(perm,:)*(T(perm,:)'*x))));
    [yp,sr2p] = eigs(MVfcn,m,1);
    rp = yp'*T(perm,:);
    Dr2(t,:) = rp.^2; Dsr2(t)=sr2p;
end
p=zeros(d,1);                % p-values computation
for k=1:d
    p(k) = sum(double(Dr2(1,k)<=Dr2(:,k)))/np;
end
psr2 = sum(double(Dsr2(1)<=Dsr2))/np;
if fwerc>0
    s = Holm(p,a);
else
    s = double(p<=a);
end
end


function S=Holm(P,a)
% Holm-Bonferroni control of the Family-Wise Error Rate
% P: matrix of p-values; a: chosen alpha risk
% S: matrix of significance (1= significant, 0= n.s.)
d=size(P); P=P(:); n=length(P);
[P,I]=sort(P); h=a./(n-(1:n)'+1);
k=find(P>h,1,'first');
if isempty(k)
    S=ones(n,1);
else if k==1
        S=zeros(n,1);
    else
        S(I)=double((1:n)'<k);
    end
end
S=reshape(S,d);
end


function [B,mse]=MTRegLS(Xi,Th,w)
% Compute the least-squares solution associated with the
%    output of the MTRegPV function
%             Input arguments:
% Xi: m x n matrix of regressors
% Th: m x d matrix of target variables
% w: n x 1 vector of regression coefficients provided
%    by the function MTRegPV(Xi,Th)
%             Output arguments:
% B: (n+1)x d matrix of rank-one regression coefficients
```

```
% mse: global mean squared residual error
% ------------------------------------------------------
[m,n]=size(Xi); [m1,d]=size(Th);
if m~=m1, error('Different numbers of items'); end
mux=mean(Xi); mut=mean(Th);
X=Xi-ones(m,1)*mux; Y=Th-ones(m,1)*mut;
y=X*w; v=y'*Y/(y'*y);
B=w*v; mu=mut-mux*B; B=[mu;B];
ThA=[ones(m,1),Xi]*B;                % Th approximation
er2=(Th-ThA).^2; mse=mean(er2(:));   % Residual error
end
```

Random example of use:

```
>> m=10; n=5; d=4; X=rand(m,n); T=rand(m,d); fwerc=0; a=0.05;

>> [sr2,psr2,r,p,s,w] = MTRegPV(X,T,fwerc,a);

>> [sr2,psr2]

ans =

   0.9600    0.7470

>> [r,p,s]

ans =

  -0.2784    0.7950         0

   0.9015    0.0100    1.0000

   0.2301    0.8540         0

   0.1299    0.8260         0

>> w'

ans =

   0.0539   -0.4747    0.7709   -0.5842   -0.9033
```
