2016 IEEE Statistical Signal Processing Workshop (SSP)

SPARSE MULTIVARIATE FACTOR REGRESSION

Milad Kharratzadeh and Mark Coates

Department of Electrical and Computer Engineering, McGill University

ABSTRACT

We introduce a sparse multivariate regression algorithm which simultaneously performs dimensionality reduction and parameter estimation. We decompose the coefficient matrix into two sparse matrices: a long matrix mapping the predictors to a set of factors and a wide matrix estimating the responses from the factors. We impose an elastic net penalty on the former and an ℓ_1 penalty on the latter. Our algorithm simultaneously performs dimension reduction and coefficient estimation and automatically estimates the number of latent factors from the data. Our formulation results in a non-convex optimization problem, which despite its flexibility to impose effective low-dimensional structure, is difficult, or even impossible, to solve exactly in a reasonable time. We specify a greedy optimization algorithm based on alternating minimization to solve this non-convex problem and provide theoretical results on its convergence and optimality. Finally, we demonstrate the effectiveness of our algorithm via experiments on simulated and real data.

Index Terms— Sparse Multivariate Regression, Factor Regression, Low Rank, Sparse Principal Component Analysis

1. INTRODUCTION

Multivariate Regression. We study the problem of linear multivariate regression where we have a set of p-dimensional predictors, $\mathbf{x_i} = (x_{i1}, \ldots, x_{ip})^T \in \mathbb{R}^p$, and q-dimensional responses, $\mathbf{y_i} = (y_{i1}, \ldots, y_{iq})^T \in \mathbb{R}^q$, which are related as:

$$\mathbf{y_i} = \mathbf{D}^T\mathbf{x_i} + \boldsymbol{\epsilon}_i, \qquad i = 1, \ldots, n, \tag{1}$$

where $\mathbf{D}_{p\times q}$ is the regression coefficient matrix and $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{iq})$ is the vector of errors for the i-th sample. We assume that the error vectors for N samples are i.i.d. Gaussian random vectors with zero mean and covariance $\boldsymbol{\Sigma}$, i.e. $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), i = 1, \ldots, N$. Rewriting (1) in matrix format, we have:

$$\mathbf{Y} = \mathbf{XD} + \mathbf{E}, \tag{2}$$

where $\mathbf{X}$ denotes the $n \times p$ matrix of predictors with $\mathbf{x_i}^T$ as its i-th row, $\mathbf{Y}$ denotes the $n \times q$ matrix of responses with $\mathbf{y_i}^T$ as its i-th row, and $\mathbf{E}$ denotes the $n \times q$ matrix of errors with $\boldsymbol{\epsilon}_i^T$ as its i-th row. In this paper, it is assumed that the columns of $\mathbf{X}$ and $\mathbf{Y}$ are centred and thus, the intercept term is removed. In this paper, we are interested in multivariate regression tasks where it is reasonable to believe that the responses are related to factors, each of which is a sparse linear combination of the predictors. Our model further assumes that the relationships between the factors and the responses are sparse. This type of structure occurs in a number of applications and we provide two examples later.

Regularization. Standard regression techniques, such as least-squares or principal component regression, are not consistent unless $p/n \to 0$. Therefore, it is necessary to impose structural constraints on the coefficient matrix in high-dimensional settings. The most common constraint is the ℓ_1 penalty (LASSO) which imposes element-wise sparsity on the coefficient matrix [1, 2]. For the multivariate regression problem in (2), ℓ_1 regularization reduces the problem to q separate multiple regression problems being solved independently. This is not desirable since the correlation between the tasks is not used. Another way to introduce sparsity is to consider the mixed ℓ_1/ℓ_γ norms ($\gamma > 1$). This regularization (sometimes called group LASSO), imposes a block-sparse structure where each row is either all zero or mostly zeros. Particular examples, among many other works, include results using the ℓ_1/ℓ_∞ norm [3, 4], and the ℓ_1/ℓ_2 norm [5–7]. Peng et al. proposed a method, called RemMap [8], which imposes both element-wise and row-wise sparsity and solves the:

$$\min_{\mathbf{D}} \|\mathbf{Y} - \mathbf{XD}\|_F^2 + \lambda_1\|\mathbf{D}\|_{1,1} + \lambda_2\|\mathbf{D}\|_{1,2}.$$

In an alternative approach, [9] extended the partial least squares (PLS) framework by imposing an additional sparsity constraint and proposed Sparse PLS (SPLS).

Factor Regression. Instead of directly regularizing the regression coefficients, one can impose constraints on the rank of the coefficient matrix, its singular values and/or its singular vectors [9–13]. These algorithms belong to a broad family of dimension-reduction methods known as *linear factor regression*, where the responses are regressed on a set of *factors* achieved by a linear transformation of the predictors. Thus, in factor regression, the coefficient matrix is decomposed into two matrices $\mathbf{D} = \mathbf{A}_{p\times m}\mathbf{B}_{m\times q}$, where $\mathbf{A}$ transforms the predictors into m latent factors, and $\mathbf{B}$ determines the factor loadings.

Our contributions. Here, we propose a novel algorithm which performs sparse multivariate factor regression (SMFR). We jointly estimate matrices $\mathbf{A}$ and $\mathbf{B}$ by minimizing the mean-squared error, $\|\mathbf{Y} - \mathbf{XAB}\|_F^2$, with an elastic net penalty on $\mathbf{A}$ (which promotes grouping of correlated predictors and the interpretability of the factors) and an ℓ_1 penalty on $\mathbf{B}$ (which enhances the accuracy and interpretability of the predictions). We provide a formulation to estimate the number of effective latent factors, m. To the best of our knowledge, our work is the first to strive for low-dimensional structure by imposing sparsity on both factoring and loading matrices as well as the grouping of the correlated predictors. This can result in a set of interpretable factors and loadings with high predictive power; however, these benefits come at the cost of a non-convex objective function. Most current approaches for multivariate regression solve a convex problem (either through direct formulation or by relaxation of a non-convex problem) to impose low-dimensional structures on the coefficient matrix. Although non-convex formulations, such as the one introduced here, can be employed to achieve very effective representations in the context of multivariate regression, there are

978-1-4673-7802-4/16/$31.00 ©2016 IEEE

few theoretical performance guarantees for optimization schemes solving such problems. We formulate our problem in Section 2. In Section 3, we propose an alternating minimization scheme to solve our problem and provide theoretical guarantees for its convergence and optimality. We show that under mild conditions on the predictor matrix, every limit point of the minimization algorithm is a stationary point of the objective function and if the starting point is close enough to a local or global minimum, our algorithm converges to that point. Finally, through analysis of simulations on synthetic and real datasets in Sections 4 and Section 5, we show that compared to other multivariate regression algorithms, our proposed algorithm can provide a more effective representation of the data, resulting in a higher predictive power.

Related Work. Our algorithm belongs to the class of low-rank multivariate regression algorithms. A well-known algorithm in this class is an approach called Reduced Rank Regression (RRR) [13] where the sum-of-squares error is minimized under the constraint that $\text{rank}(\mathbf{D}) \le m$ for some $m \le \min\{p, q\}$. Without regularization, RRR has poor predictive performance and is not suitable for high-dimensional settings. To resolve this, it is common to use the trace norm as the penalty [12, 14–16]: $\min_{\mathbf{D}} \|\mathbf{Y} - \mathbf{X}\mathbf{D}\|_F^2 + \lambda \sum_{j=1}^{\min\{p,q\}} \sigma_j(\mathbf{D})$, where $\sigma_j(\mathbf{D})$ denotes the j'th singular value of $\mathbf{D}$. By imposing sparsity in the singular values of $\mathbf{D}$, trace norm regularization results in a low-dimensional solution. Another common formulation is: $\min_{\mathbf{D}} \|\mathbf{Y}{-}\mathbf{X}\mathbf{D}\|_F^2 + g(\mathbf{D})$ s.t. $\text{rank}(\mathbf{D}) \le m$, where $g(\mathbf{D})$ is a regularization function over $\mathbf{D}$. For instance, in [17], a ridge penalty is proposed with $g(\mathbf{D}) = \lambda\|\mathbf{D}\|_F^2$. Often, the rank constraint on $\mathbf{D}$ is enforced by assuming that $\mathbf{D} = \mathbf{A}_{p\times m}\mathbf{B}_{m\times q}$ and the problem is formulated in terms of $\mathbf{A}$ and $\mathbf{B}$. In [18], $g(\mathbf{A}, \mathbf{B}) = \lambda_1\|\mathbf{A}\|_{1,1} + \lambda_2\|\mathbf{B}\|_F^2$. An algorithm called Sparse Reduced Rank Regression (SRRR) is proposed in [11], where $g(\mathbf{A}, \mathbf{B}) = \lambda\|\mathbf{A}\|_{1,2}$ with an additional constraint that $\mathbf{B}\mathbf{B}^T = \mathbf{I}$. In [14], $g(\mathbf{A}, \mathbf{B}) = \lambda\|\mathbf{B}\|_{2,1}^2$ with an extra constraint that $\mathbf{A}^T\mathbf{A} = \mathbf{I}$, and it is assumed that $m = p \le q$. The constraint on $\mathbf{B}$ forces many rows to be zero which cancels the effects of the corresponding columns in $\mathbf{A}$.

Our problem formulation differs in three important ways: (i) sparsity constraints are imposed on *both* $\mathbf{A}$ and $\mathbf{B}$; (ii) the elastic net penalty on $\mathbf{A}$ provides the grouping of correlated predictors; and (iii) the number of factors is determined directly, without the need for cross-validation. We will discuss the second and third aspects in detail in the next section. The first difference has substantial consequences; when decomposing the coefficient matrix into two matrices, the first matrix has the role of aggregating the input signals to form the latent factors and the second matrix performs a multivariate regression on these factors. Imposing sparsity on $\mathbf{A}$ enhances the variable selection as well as the interpretability of the achieved factors. Also, as originally motivated by LASSO, we would like to impose the sparsity constraint on $\mathbf{B}$ in order to improve the interpretability and prediction performance.

2. PROBLEM SETUP

In this work, we introduce a novel low-dimensional structure where we decompose $\mathbf{D}$ into the product of two sparse matrices $\mathbf{A}_{p\times m}$ and $\mathbf{B}_{m\times q}$ where $m < \min(p, q)$. This decomposition can be interpreted as first identifying a set of m factors which are derived by some linear transformation of the predictors (through matrix $\mathbf{A}$) and then identifying the transformed regression coefficient matrix $\mathbf{B}$ to estimate the responses from these m factors. We provide a framework to find m, the number of effective latent factors, as well as matrices $\mathbf{A}$ and $\mathbf{B}$. For a fixed m, define:

$$\widehat{\mathbf{A}}_m, \widehat{\mathbf{B}}_m = \underset{\mathbf{A}_{p\times m}, \mathbf{B}_{m\times q}}{\arg\min}\; f(\mathbf{A}, \mathbf{B}), \tag{3}$$

where

$$f(\mathbf{A}, \mathbf{B}) = \frac{1}{2}\|\mathbf{Y}{-}\mathbf{X}\mathbf{A}\mathbf{B}\|_F^2 + \lambda_1\|\mathbf{A}\|_{1,1} + \lambda_2\|\mathbf{B}\|_{1,1} + \lambda_3\|\mathbf{A}\|_F^2. \tag{4}$$

Then, we solve the following optimization problem:

$$\widehat{m} = \max(m) \le r \;\text{ s.t. }\; \text{rank}(\widehat{\mathbf{A}}_m) = \text{rank}(\widehat{\mathbf{B}}_m) = m, \tag{5}$$

where r is a problem-specific bound on the number of factors. We then choose $\widehat{\mathbf{A}}_{\widehat{m}}$ and $\widehat{\mathbf{B}}_{\widehat{m}}$ as solutions. Thus, we find the maximum number of factors such that the solution of (3) has full rank factor and loading matrices. In other words, we find the maximum m such that the best possible regularized reconstruction of responses, i.e., the solution of (3), results in a model where the factors (columns of of $\widehat{\mathbf{A}}$) and their contributions to the responses (rows of $\widehat{\mathbf{B}}$) are linearly independent. To achieve this, we initialize $\mathbf{A}$ to have r columns, $\mathbf{B}$ to have r rows, and set $m = r$, solve the problem (3)–(4), check for the full rank condition; if not satisfied, set $m = m - 1$, and repeat the process until we identify an m that satisfies the rank condition.

2.1. Grouping of Correlated Features

In this section, we show the i'th row of a matrix $\mathbf{X}$ by $\mathbf{X}_{i\cdot}$ and its j'th column by $\mathbf{X}_{\cdot j}$. Remember that matrix $\mathbf{A}$ has the role of combining relevant features to form the latent factors which will be used later in the second layer by matrix $\mathbf{B}$ for estimating the outputs. If there are two highly correlated features we expect them to be grouped together in forming the factors. In other words, we expect them to be both present in a factor or both not present. Inspired by Theorem 1 in the original paper of Zou and Hastie on elastic net [19], we prove in this section that elastic net penalty enforces the grouping of correlated features in forming the factors. The columns of $\mathbf{X}$ correspond to different features. We assume that all columns of $\mathbf{X}$ are centred and normalized. Thus, the correlation between the i'th and the j'th features is $\rho_{ij} \triangleq \mathbf{X}_{\cdot i}^T\mathbf{X}_{\cdot j}$.

Lemma 1. *Consider solving* $\widehat{\mathbf{A}} = \arg\min_{\mathbf{A}} f(\mathbf{A}, \mathbf{B})$. *Then, if* $\widehat{\mathbf{A}}_{ik}\widehat{\mathbf{A}}_{jk} > 0$, *we have:*

$$\frac{2\lambda_3}{\|\mathbf{Y}\|_F\|\mathbf{B}_{k\cdot}\|_F}|\widehat{\mathbf{A}}_{ik} - \widehat{\mathbf{A}}_{jk}| \le \sqrt{2(1-\rho_{ij})} \tag{6}$$

The proof is omitted due to lack of space, but can be found in the extended version of this paper [20]. This lemma says, for instance, that if the correlation between features i and j is really high (i.e., $\rho_{ij} \approx 1$), then the difference between their corresponding weights in forming the k'th factor, $|\widehat{\mathbf{A}}_{ik} - \widehat{\mathbf{A}}_{jk}|$, would be very close to 0. If $\mathbf{X}_{\cdot i}$ and $\mathbf{X}_{\cdot j}$ are negatively correlated, we can state the same lemma for $\mathbf{X}_{\cdot i}$ and $-\mathbf{X}_{\cdot j}$ and use $|\rho_{ij}|$.

2.2. Estimating the number of effective factors

In choosing m, we want to avoid both overfitting (large m) and lack of sufficient learning power (small m). In general, we only require $m \le \min(p, q)$; however, in practical settings where p and q are very large, we impose an upper bound on m to have a reasonable number of factors and avoid overfitting. This upper bound, denoted by r, is problem-specific and should be chosen by the programmer. In order to have the maximum learning power, we find the maximum

$m \leq r$ for which the solutions satisfy our rank conditions. This motivates starting with $m = r$ and decreasing it until the conditions hold. The full rank conditions are employed to guarantee a good estimate of the number of "effective" factors. An effective factor explains some aspect of the response data but cannot be constructed as a linear combination of other factors. We therefore require the estimated factors to be linearly independent. In addition, we require that the rows of $\mathbf{B}$, which determine how the factors affect the responses, are linearly independent. If we do not have this latter independence, we could reduce the number of factors and still obtain the same relationship matrix $\mathbf{D}$, so at least one of the factors is superfluous. By enforcing that $\mathbf{A}$ and $\mathbf{B}$ are full rank, we make sure that the estimated factors are linearly independent in both senses, and thus $\widehat{m}$ is a good estimate of the number of effective factors.

3. OPTIMIZATION AND THEORETICAL RESULTS

The optimization problem defined in (3–5) is not a convex problem and it is difficult, if not impossible, to solve exactly (i.e., to find the global optimum) in polynomial time. Therefore, we have to employ heuristic algorithms, which may or may not converge to a stationary solution [21]. In this section, we propose an alternating minimization algorithm and provide some theoretical results.

For a fixed m, the objective function in (3) is biconvex in $\mathbf{A}$ and $\mathbf{B}$; it is not convex in general, but is convex if either $\mathbf{A}$ or $\mathbf{B}$ is fixed. Let us define $\mathbf{C} = (\mathbf{A}, \mathbf{B})$. To solve (3) for a fixed m, we perform the following alternating updates with an arbitrary, non-zero starting value $\mathbf{C}_0 = (\mathbf{A}_0, \mathbf{B}_0)$:

$$\mathbf{B}_{i+1} \leftarrow \arg\min_{B} \frac{1}{2}\|\mathbf{Y} - \mathbf{X}\mathbf{A}_i\mathbf{B}\|_F^2 + \lambda_2\|\mathbf{B}\|_{1,1} \tag{7}$$

$$\mathbf{A}_{i+1} \leftarrow \arg\min_{A} \frac{1}{2}\|\mathbf{Y}-\mathbf{X}\mathbf{A}\mathbf{B}_{i+1}\|_F^2 + \lambda_3\|\mathbf{A}\|_F^2 + \lambda_1\|\mathbf{A}\|_{1,1} \tag{8}$$

The stopping criterion is related to the convergence of the value of function f, not the convergence of its arguments. In our experiments, we assume f has converged, if $|f_i - f_{i+1}|/f_i < \epsilon$ where the default value of the tolerance parameter, ϵ, is $1E-5$. It is possible to modify the updates in (7) and (8) to achieve faster convergence; we do not include these results here due to lack of space. These results as well as all the proofs for the Theorems in this section can be found in the extended version of this paper [20].

Definition 1. $\mathbf{C}^* = (\mathbf{A}^*, \mathbf{B}^*)$ is called a *partial optimum* of f if

$$\begin{aligned} f(\mathbf{A}^*, \mathbf{B}^*) &\leq f(\mathbf{A}^*, \mathbf{B}),\ \forall \mathbf{B} \in \mathbb{R}^{m\times q} & (9)\\ \text{and} \quad f(\mathbf{A}^*, \mathbf{B}^*) &\leq f(\mathbf{A}, \mathbf{B}^*),\ \forall \mathbf{A} \in \mathbb{R}^{n\times m}. & (10)\end{aligned}$$

Definition 2. A point $\mathbf{C}^*$ is an *accumulation point* or a *limit point* of a sequence $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$, if for any neighbourhood V of $\mathbf{C}^*$, there are infinitely many $j \in \mathbb{N}$ such that $\mathbf{C}_j \in V$. Equivalently, $\mathbf{C}^*$ is the limit of a subsequence of $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$.

Proposition 1. *The sequence $f(\mathbf{A}_i, \mathbf{B}_i)$ generated by Algorithm 1 converges monotonically.*

The value of f is always positive and is reduced in each of the two main steps of Algorithm 1. Thus, it is guaranteed that the stopping criterion of Algorithm 1 will be reached.

Theorem 1. *If the entries of $\mathbf{X} \in \mathbb{R}^{n\times p}$ are drawn from a continuous probability distribution on $\mathbb{R}^{np}$, then: (i) The solution of (7) is unique if $\mathbf{A}_i$ is full rank. (ii) The objective of (8) is strongly convex and its solution, if one exists, is unique.*

In classical LASSO, the condition on the entries of $\mathbf{X}$ is sufficient to achieve solution uniqueness [22]. For LASSO, the continuity is used to argue that the columns of $\mathbf{X}$ are in *general position* with probability 1. The affine span of the columns of $\mathbf{X}$, $\{\mathbf{X}_1, \dots, \mathbf{X}_{k+1}\}$, has Lebesgue measure 0 in $\mathbb{R}^n$ for a continuous distribution on $\mathbb{R}^n$, so there is zero probability of drawing $\mathbf{X}_{k+2}$ in their span. If we multiply $\mathbf{X}$ by a matrix with full column rank, we retain the same property, and thus the solution of (7) is unique if $\mathbf{A}_i$ is full rank. Since the objective function is strictly convex in $\mathbf{A}$ (due to the elastic net property), if (8) has a solution, its solution is unique. Next, we study the properties of $\widehat{\mathbf{C}} = (\widehat{\mathbf{A}}, \widehat{\mathbf{B}})$ at convergence.

Theorem 2. *Assume that the entries of $\mathbf{X} \in \mathbb{R}^{n\times p}$ are drawn from a continuous probability distribution on $\mathbb{R}^{np}$. For a given starting point $\mathbf{A}_0$, let $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$ denote the sequence of solutions generated by Algorithm 1. Then: (i) $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$ has at least one accumulation point. (ii) All the accumulation points of $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$ are partial optima and have the same function value. (iii) If $\mathbf{B}$ is full rank for all accumulation points of $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$, then:*

$$\lim_{i\to\infty} \|\mathbf{C}_{i+1} - \mathbf{C}_i\| = 0, \tag{11}$$

Part (i) follows from the fact that the solutions produced by Algorithm 1 are contained in a bounded, closed (and hence compact) set. Although Algorithm 1 converges to a specific value of f, this value can be achieved by different values of $\mathbf{C}$. Thus, the sequence $\mathbf{C}_i$ can have many accumulation points. Part (ii) of Theorem 2 shows that any accumulation point is a partial optimum. Proposition 1 implies that for any given starting point, all the associated accumulation points have the same f value. Under the assumption that $\mathbf{B}$ is full rank for all accumulation points of $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$, part (iii) provides a guarantee that the difference between successive solutions of the algorithm converges to zero, for both the factor and loading matrices. Although the condition in (11) does not guarantee the convergence of the sequence $\{\mathbf{C}_i\}_{i\in\mathbb{N}}$, it is close enough for practical purposes. Also, note that for finding the number of factors, we require that both $\mathbf{A}$ and $\mathbf{B}$ to be full rank for the final solution. When $\mathbf{B}$ is full rank, the solutions to both (7) and (8) are unique and thus $\mathbf{A}$ and $\mathbf{B}$ will not change in the following iterations, i.e., convergence.

4. SIMULATION STUDY

In this section, we use synthetic data to compare the performance of our algorithm with several related multivariate regression methods reviewed in the introduction. We generate the synthetic data in accordance with the model described in (2), $\mathbf{Y} = \mathbf{X}\mathbf{D} + \mathbf{E}$, where $\mathbf{D} = \mathbf{A}\mathbf{B}$. First, we generate an $n \times p$ predictor matrix, $\mathbf{X}$, with rows independently drawn from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_X)$, where the (i,j)-th element of $\mathbf{\Sigma}_X$ is defined as $\sigma^X_{i,j} = 0.7^{|j-i|}$. This is a common model for predictors in the literature [8,12,23]. The rows of the $n \times q$ error matrix are sampled from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_N)$, where the (i,j)-th element of $\mathbf{\Sigma}_N$ is defined as $\sigma^X_{i,j} = \sigma_n^2 \cdot 0.4^{|j-i|}$. The value of σ_n^2 is varied to attain different levels of signal to noise ratio (SNR). Each row of the $p \times m$ matrix $\mathbf{A}$ is chosen by first randomly selecting m_0 of its elements and sampling them from $\mathcal{N}(0,1)$ and then setting the rest of its elements to zero. Finally, we generate the $m \times q$ matrix $\mathbf{B}$ by the element-wise product of $\mathbf{B} = \mathbf{U} \circ \mathbf{W}$, where the elements of $\mathbf{U}$ are drawn independently from $\mathcal{N}(0,1)$ and elements of $\mathbf{W}$ are drawn from Bernoulli distribution with success probability s. We evaluate the predictive performance over a separate test set $(\mathbf{X}_{test}, \mathbf{Y}_{test})$, in terms of the mean-squared error: MSE $= \|\mathbf{X}_{test}\widehat{\mathbf{D}} - \mathbf{Y}_{test}\|_F^2/nq$, where $\widehat{\mathbf{D}}$ is the estimated coefficient matrix. In our case, $\widehat{\mathbf{D}} = \widehat{\mathbf{A}}\widehat{\mathbf{B}}$.

Parameters							MSE over test set							
n	p	q	m	m_0	σ_n	s	SMFR	LASSO	ℓ_1/ℓ_2 [6]	SRRR [11]	RemMap [8]	SPLS [9]	Trace [15]	Ridge
50	150	50	10	1	3	0.2	0.070	0.083	0.090	0.084	0.083	0.091	0.088	0.089
							(0.004)	(0.005)	(0.005)	(0.005)	(0.005)	(0.007)	(0.004)	(0.004)
			10	1	3	0.4	0.078	0.104	0.105	0.099	0.104	0.110	0.110	0.111
							(0.007)	(0.008)	(0.007)	(0.007)	(0.008)	(0.008)	(0.007)	(0.006)
			10	1	5	0.2	0.110	0.118	0.133	0.117	0.123	0.122	0.115	0.122
							(0.004)	(0.005)	(0.004)	(0.005)	(0.004)	(0.007)	(0.005)	(0.006)
			15	2	3	0.2	0.071	0.108	0.112	0.109	0.107	0.114	0.109	0.110
							(0.003)	(0.006)	(0.007)	(0.008)	(0.006)	(0.008)	(0.006)	(0.008)
50	100	100	10	1	5	0.1	0.068	0.070	0.092	0.071	0.075	0.073	0.071	0.074
							(0.001)	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
500	150	50	10	1	3	0.2	0.0172	0.0180	0.0198	0.0176	0.0184	0.0216	0.0183	0.0187
							(0.0001)	(0.0001)	(0.0001)	(0.0001)	(0.0001)	(0.0007)	(0.0002)	(0.0001)
500	100	100	10	1	5	0.3	0.0202	0.0209	0.0222	0.0204	0.0214	0.0222	0.0208	0.0213
							(0.0001)	(0.0002)	(0.0001)	(0.0001)	(0.0001)	(0.0003)	(0.0001)	(0.0002)

Table 1: Comparison of six algorithms for different setups. We report mean and standard deviations of the MSE over the test sets (20 runs).

We compare the performance of our algorithm, SMFR, with many other algorithms reviewed in Section 1 as well as a baseline algorithm with a simple ridge penalty. The means and standard deviations of different algorithms are presented in Table 1. We use five-fold cross-validation to find the tuning parameters of all algorithms. We set r, the maximum number of factors, to 20. Our algorithm outperforms the other algorithms and results in lower MSE means and standard deviation. On average, our algorithm reduces the test error by 13.2% compared with LASSO, 21.4% compared with ℓ_1/ℓ_2, 12.3% compared with SRRR, 15.2% compared with RemMap, 19.4% compared with SPLS, 39.1% compared with Trace, and 16.7% compared with Ridge. See [20] for more results.

5. APPLICATION TO REAL DATA

We consider a dataset of Montreal's bicycle sharing system called BIXI. The data contains the number of available bikes in each of the 400 installed stations for every minute. We use the data collected for the first four weeks of June 2012. We allocate two features to each station corresponding to the number of arrivals and departures of bikes to or from that station for every hour. The learning task is to predict the number of arrivals and departures for all the stations from the number of arrivals and departures in the last hour (i.e., a vector autoregressive model). We perform the prediction task on each of the four weeks. For each week, we take the data for the first 5 days (Friday to Tuesday) as the training set, and the last two days as the test set (Wednesday and Thursday). We compare the algorithms performing dimensionality reduction in terms of their predictive performance on the test sets in Table 2. We also include LASSO and ℓ_1/ℓ_2 as baseline algorithms. We observe that our algorithm outperforms the others in all 4 weeks.

week	SMFR	SRRR	SPLS	LASSO	ℓ_1/ℓ_2
1	557.3	570.0	1661	580.4	591.0
2	570.1	602.2	1888	610.9	623.7
3	618.8	641.9	2159	643.4	657.8
4	549.7	594.6	1621	594.9	588.0

Table 2: Total squared error (MSE×nq) for BIXI dataset

We examine one of the factors built by our algorithm in Fig. 1. With no geographical data provided to the algorithm, this learned factor shows that the departures from populated residential areas (The Plateau, Mile End, Outremont) and arrivals at downtown (Ville Marie) are combined together to form a factor. This agrees with the intuition that many people are taking bikes to go from their homes to downtown where universities and businesses are located and shows the success of our algorithm in discovering the underlying structure of the data which in turn results in better predictive performance.

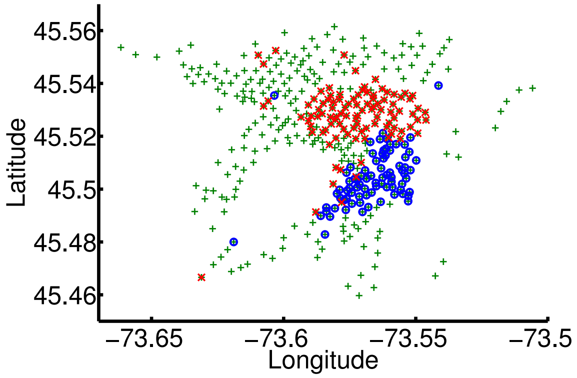

Fig. 1: One of the factors identified in the BIXI dataset by our algorithm. Green plus signs show the stations, red crosses show the departure features and blue circles show the arrival features.

6. CONCLUSION

We introduced a sparse multivariate regression algorithm which imposes a novel sparse and low-dimensional structure on the coefficient matrix and promotes grouping of correlated features. We also provided a formulation to infer the number of latent factors in a more effective way than current techniques. Although the problem formulation leads to a non-convex optimization problem, we showed that the proposed alternating minimization scheme converges to a partial optimum. Through experiments on simulated and real datasets, we demonstrated that the proposed algorithm is able to exploit the existing structure in the data to improve predictive performance.

7. REFERENCES

[1] Robert J. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," *J. Royal Statistical Society: Series B*, vol. 73, no. 3, pp. 273–282, 2011.

[2] Martin J Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (LASSO)," *IEEE Trans. Info. Theory*, vol. 55, no. 5, pp. 2183–2202, 2009.

[3] Berwin A. Turlach, William N. Venables, and Stephen J. Wright, "Simultaneous variable selection," *Technometrics*, vol. 47, no. 3, pp. 349–363, 2005.

[4] Cun-Hui Zhang and Jian Huang, "The sparsity and bias of the lasso selection in high-dimensional linear regression," *The Annals of Statistics*, vol. 36, no. 4, pp. 1567–1594, 2008.

[5] Ming Yuan and Yi Lin, "Model selection and estimation in regression with grouped variables," *J. Royal Statistical Society, Series B*, vol. 68, no. 1, pp. 49–67, 2006.

[6] Guillaume Obozinski, Martin J Wainwright, Michael I Jordan, et al., "Support union recovery in high-dimensional multivariate regression," *The Annals of Statistics*, vol. 39, no. 1, pp. 1–47, 2011.

[7] Xiaolei Lv, Guoan Bi, and Chunru Wan, "The group lasso for stable recovery of block-sparse signal representations," *IEEE Trans. Signal Processing*, vol. 59, no. 4, pp. 1371–1382, 2011.

[8] Jie Peng, Ji Zhu, Anna Bergamaschi, Wonshik Han, Dong-Young Noh, Jonathan R. Pollack, and Pei Wang, "Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer," *The Annals of Applied Statistics*, vol. 4, no. 1, pp. 53–77, 2010.

[9] Hyonho Chun and Sündüz Keleş, "Sparse partial least squares regression for simultaneous dimension reduction and variable selection," *J. Royal Statistical Society: Series B*, vol. 72, no. 1, pp. 3–25, 2010.

[10] M. Pourahmadi, *High-Dimensional Covariance Estimation: With High-Dimensional Data*, Wiley Series in Probability and Statistics. Wiley, 2013.

[11] Lisha Chen and Jianhua Z Huang, "Sparse reduced-rank regression for simultaneous dimension reduction and variable selection," *J. American Statistical Association*, vol. 107, no. 500, pp. 1533–1545, 2012.

[12] Ming Yuan, Ali Ekici, Zhaosong Lu, and Renato Monteiro, "Dimension reduction and coefficient estimation in multivariate linear regression," *J. Royal Statistical Society, Series B*, vol. 69, no. 3, pp. 329–346, 2007.

[13] Raja Velu and Gregory C Reinsel, *Multivariate reduced-rank regression: theory and applications*, Springer, 1998.

[14] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil, "Convex multi-task feature learning," *Machine Learning*, vol. 73, no. 3, pp. 243–272, 2008.

[15] Shuiwang Ji and Jieping Ye, "An accelerated gradient method for trace norm minimization," in *Proc. Int. Conf. Machine Learning*. ACM, 2009, pp. 457–464.

[16] Xinhua Zhang, Dale Schuurmans, and Yao-liang Yu, "Accelerated training for matrix-norm regularization: A boosting approach," in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 2906–2914.

[17] Ashin Mukherjee and Ji Zhu, "Reduced rank ridge regression and its kernel extensions," *Statistical Analysis and Data Mining*, vol. 4, no. 6, pp. 612–622, 2011.

[18] Abhishek Kumar and Hal Daume, "Learning task grouping and overlap in multi-task learning," *arXiv:1206.6417*, 2012.

[19] Hui Zou and Trevor Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B*, vol. 67, no. 2, pp. 301–320, 2005.

[20] Milad Kharratzadeh and Mark Coates, "Sparse multivariate factor regression," 2015, Technical Report, McGill University, available at http://networks.ece.mcgill.ca/pubs.

[21] Michael JD Powell, "On search directions for minimization algorithms," *Mathematical Programming*, vol. 4, no. 1, pp. 193–201, 1973.

[22] Ryan J Tibshirani, "The lasso problem and uniqueness," *Electronic J. Statistics*, vol. 7, pp. 1456–1490, 2013.

[23] Adam J. Rothman, Elizaveta Levina, and Ji Zhu, "Sparse multivariate regression with covariance estimation," *J. Computational and Graphical Statistics*, vol. 19, no. 4, pp. 947–962, 2010.