



LDR: A Package for Likelihood-Based Sufficient Dimension Reduction

R. Dennis Cook
University of Minnesota

Liliana M. Forzani
IMAL

Diego R. Tomassi
Universidad Nacional del Litoral

Abstract

We introduce a new MATLAB software package that implements several recently proposed likelihood-based methods for sufficient dimension reduction. Current capabilities include estimation of reduced subspaces with a fixed dimension d , as well as estimation of d by use of likelihood-ratio testing, permutation testing and information criteria. The methods are suitable for preprocessing data for both regression and classification. Implementations of related estimators are also available. Although the software is more oriented to command-line operation, a graphical user interface is also provided for prototype computations.

Keywords: dimension reduction, inverse regression, principal components.

1. Introduction

Many statistical problems deal with the study of the conditional distribution of a response Y given a vector of predictors $\mathbf{X} \in \mathbb{R}^p$ or with the relationship among a set of variables \mathbf{X} in different populations $Y = 1, \dots, h$. When p is large, almost all of the methods used to study these relationships include some type of dimension reduction for \mathbf{X} . Principal components is perhaps the dominant method of dimension reduction across the applied sciences, although there are other established and recent statistical methods that can be used in these regression settings, including partial least squares (Wold 1996; Helland 1990), projection pursuit (Huber 1985), and sparse methods based on the lasso (Tibshirani 1996). A new paradigm – called *sufficient dimension reduction* (SDR) – has developed over the past two decades for studying intrinsically low-dimensional regressions, often without requiring a pre-specified model. The following definition provides a conceptual foundation for SDR and for the models and methods for which we developed software tools.

Definition: (Cook 2007) A reduction $\mathbf{R} : \mathbb{R}^p \rightarrow \mathbb{R}^q$, $q \leq p$, is sufficient for $Y|\mathbf{X}$ if it satisfies

one of the following three statements:

- (i) Inverse reduction, $\mathbf{X}|(Y, \mathbf{R}(\mathbf{X})) \sim \mathbf{X}|\mathbf{R}(\mathbf{X})$
- (ii) Forward reduction, $Y|\mathbf{X} \sim Y|\mathbf{R}(\mathbf{X})$
- (iii) Joint reduction, $\mathbf{X} \perp\!\!\!\perp Y|\mathbf{R}(\mathbf{X})$

where $A|B$ refers to a random quantity A given a second random quantity B , $\perp\!\!\!\perp$ indicates independence and \sim means identically distributed. The choice of a reductive paradigm depends on the stochastic nature of \mathbf{X} and Y . If the values of \mathbf{X} are fixed by design, then forward regression (ii) seems the natural choice. In discriminant analysis $\mathbf{X}|Y$ is a random vector of features observed in one of a number of populations indexed by Y . In this case inverse regression (i) is perhaps the only reasonable reductive route. The three statements in the definition are equivalent when Y and \mathbf{X} have a joint distribution (iii). In this last case we are free to estimate a reduction inversely or jointly and then pass the estimated reduction to the forward regression without additional structure.

Among the methods that fall under the SDR paradigm we can include sliced inverse regression (SIR, Li 1991), sliced average variance estimation (SAVE, Cook and Weisberg 1991), parametric inverse regression (PIR, Bura and Cook 2001), minimum average variance estimation (MAVE, Xia, Tong, Li, and Zhu 2002), contour regression (Li, Zha, and Chiaromonte 2005), inverse regression estimation (Cook and Ni 2005), Fourier methods (Zhu and Zheng 2006), and a variety of methods based on marginal moments like principal Hessian directions (Li 1992), iterative Hessian transformations (Cook and Li 2002), and directional regression (DR, Li and Wang 2007).

Nearly all SDR methods restrict attention to the class of linear reductions, which can be represented conveniently in terms of the projection $\mathbf{P}_{\mathcal{S}}\mathbf{X}$ of \mathbf{X} onto a subspace $\mathcal{S} \subseteq \mathbb{R}^p$. If $Y \perp\!\!\!\perp \mathbf{X}|\mathbf{P}_{\mathcal{S}}\mathbf{X}$ then \mathcal{S} is called a dimension-reduction subspace. Under mild conditions the intersection of any two dimension reduction subspaces is again a dimension-reduction subspace and that being so the central subspace $\mathcal{S}_{Y|\mathbf{X}}$, defined as the intersection of all dimension-reduction subspaces, is taken as the inferential target (Cook 1994, 1998). A minimal sufficient linear reduction is then of the form $\mathbf{R}(\mathbf{X}) = \boldsymbol{\alpha}^\top \mathbf{X}$, where $\boldsymbol{\alpha}$ is any basis for $\mathcal{S}_{Y|\mathbf{X}}$. Informally, SDR methods pursue the estimation of the fewest linear combinations of the predictors that contain all the regression information on the response. SIR uses a sample version of the first conditional moment $\mathbf{E}(\mathbf{X}|Y)$ to construct such an estimator, while SAVE uses sample first and second $\mathbf{E}(\mathbf{X}\mathbf{X}^\top|Y)$ conditional moments. Although SIR and SAVE have found widespread use in applications, both have well known limitations. In particular, the subspace \mathcal{S}_{SIR} estimated by SIR is typically a proper subset of $\mathcal{S}_{Y|\mathbf{X}}$ when the response surface is symmetric about the origin. SAVE was developed in response to this limitation and it provides consistent estimation of $\mathcal{S}_{Y|\mathbf{X}}$ under regularity conditions (Li and Wang 2007; Shao, Cook, and Weisberg 2007), but its ability to detect linear trends is generally inferior to SIR's. For these reasons, SIR and SAVE have been used as complementary methods, with satisfactory results often being obtained by informally combining their estimated directions (see, for example, Bura and Pfeiffer 2003; Cook 2003; Ye and Weiss 2003; Li and Li 2004; Zhu, Ohtaki, and Li 2005; Pardoe, Yin, and Cook 2007). Directional regression is a consistent estimator of $\mathcal{S}_{Y|\mathbf{X}}$ under the same conditions as SAVE.

In a series of recent articles, Cook (2007) and Cook and Forzani (2008, 2009a,b) took a substantial step forward in the development of dimension reduction methods based on the

first two conditional moments. The idea behind this new methodology is to estimate $\mathcal{S}_{Y|\mathbf{X}}$ via maximum likelihood. These likelihood-based estimators of $\mathcal{S}_{Y|\mathbf{X}}$ are consistent under the same conditions as SIR, DR and SAVE. Unlike those methods, however, a likelihood-based objective function is employed to acquire the reduced dimensions, giving these methods \sqrt{n} consistency and asymptotic efficiency at the assumed model, which is not a claimed attribute for any previous method. The dimension d of $\mathcal{S}_{Y|\mathbf{X}}$ can be estimated using likelihood ratio testing (LRT) or an information criterion like Akaike's information criterion (AIC) or Bayes information criterion (BIC), and conditional independence hypotheses involving the predictors can be tested straightforwardly. Furthermore, it was demonstrated both theoretically and with simulations that these methods have good robustness properties, even under substantial deviations of the error distribution from its nominal specification.

The goal of this article is to introduce software tools for these recently proposed likelihood-based SDR methods. We have called the package **LDR** (likelihood-based dimension reduction) and it runs under MATLAB (The MathWorks, Inc. 2009), which allows use of computational tools developed outside of the statistics community. In particular, likelihood maximization under some of the models discussed below requires optimization on a Grassmann manifold $\mathcal{G}_{(d,p)}$, which is the set of all d dimensional subspaces of \mathbb{R}^p (Edelman, Arias, and Smith 1999). Optimization on Grassmann manifolds is fairly common in some areas such as computer science and signal processing, but is relatively rare in statistics. The software is oriented to command-line operation and it is intended to provide an easy-to-use interface similar to Weisberg's **dr** package for R (Weisberg 2002, 2009). Fixed-dimension as well as dimension selection tasks are supported, both for continuous and for categorical responses. Functions to compute other estimators such as SIR, SAVE and DR are also provided.

The paper is organized as follows: In Section 2 we briefly review some theory concerning likelihood-based SDR. In Section 3, we describe the main features of the software and then in Section 4 we give examples of its use. Finally, we briefly describe how to use the graphical user interface in Section 4.5. The package and the complete software documentation are available along with this paper and from <http://liliana.forzani.googlepages.com/ldr-package>.

2. Likelihood-based SDR

The likelihood-based approach to SDR uses model-based inverse regression of \mathbf{X} on Y to gain reductive information for the forward regression of Y on \mathbf{X} (see the previous Definition). We assume that the data consist of n independent observations on (\mathbf{X}, Y) , with $\mathbf{X} \in \mathbb{R}^p$, $Y \in \mathbb{R}$, $n > p$, and that $\mathbf{X}|(Y = y) \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Delta}_y)$. Then we can write

$$\mathbf{X}_y := \mathbf{X}|(Y = y) = \boldsymbol{\mu}_y + \boldsymbol{\varepsilon}, \quad (1)$$

where $\boldsymbol{\varepsilon} \sim N(0, \boldsymbol{\Delta}_y)$, $\boldsymbol{\Delta}_y > 0$. The goal is to find the maximum likelihood estimator (MLE) for the central subspace, $\mathcal{S}_{Y|\mathbf{X}}$. Let $d = \dim(\mathcal{S}_{Y|\mathbf{X}})$ and let $\boldsymbol{\alpha}$ denote a $p \times d$ semi-orthogonal matrix whose columns form a basis for $\mathcal{S}_{Y|\mathbf{X}}$, so $Y|\mathbf{X} \sim Y|\boldsymbol{\alpha}^\top \mathbf{X}$. Likelihood-based methods are designed to estimate $\mathcal{S}_{Y|\mathbf{X}} = \text{span}(\boldsymbol{\alpha})$ under different structures for the mean $\boldsymbol{\mu}_y$ and the covariance $\boldsymbol{\Delta}_y$ functions in model (1). These methods are summarized in the sections that follow. In preparation, let $\mathcal{M} = \text{span}\{\boldsymbol{\mu}_y - \boldsymbol{\mu} | y \in S_Y\} \subseteq \mathbb{R}^p$, where S_Y denotes the sample space of Y , let $\boldsymbol{\mu} = E(\mathbf{X})$, and let \hat{a} stand for the MLE of a parameter a .

2.1. Principal components (PC)

In this model, which was introduced by Cook (2007), it is assumed that the errors are isotropic, $\Delta_y = \sigma^2 \mathbf{I}_p$. As a consequence of this structure, $\mathcal{S}_{Y|\mathbf{X}} = \mathcal{M}$ and therefore we have the representation $\mu_y = \mu + \alpha \nu_y$, where $\nu_y \in \mathbb{R}^d$ is unknown for all $y \in S_Y$. Under this setting, the MLE of $\mathcal{S}_{Y|\mathbf{X}}$ is the span of the first d eigenvectors of the sample covariance matrix $\tilde{\Sigma}$ of \mathbf{X} .

2.2. Isotropic principal fitted components (IPFC)

As in the PC model, it is assumed that the errors are isotropic. Consequently $\mathcal{S}_{Y|\mathbf{X}} = \mathcal{M}$ and we can again represent the conditional means as $\mu_y = \mu + \alpha \nu_y$. However, following (Cook 2007), the coordinate vectors ν_y are now modeled as

$$\nu_y = \beta \{\mathbf{f}_y - \mathbf{E}(\mathbf{f}_Y)\}, \quad (2)$$

where $\mathbf{f}_y \in \mathbb{R}^r$ is a known vector-valued function of y with linearly independent elements and $\beta \in \mathbb{R}^{d \times r}$, $d \leq \min(r, p)$, is an unrestricted rank d matrix. The presence of $\mathbf{E}(\mathbf{f}_Y)$ in the definition of ν_y is for ease of exposition and does not represent a restriction; only \mathbf{f}_y needs to be specified in practice. For this model the MLE of $\mathcal{S}_{Y|\mathbf{X}}$ is the span of the first d eigenvectors of $\tilde{\Sigma}_{\text{fit}}$, where $\tilde{\Sigma}_{\text{fit}}$ is the sample covariance matrix of the fitted vectors from the multivariate linear regression of \mathbf{X}_y on \mathbf{f}_y , including an intercept. When Y is categorical or slicing is used to approximate the coordinates of ν_y with step functions,

$$\tilde{\Sigma}_{\text{fit}} = \sum_{y=1}^h \frac{n_y}{n} (\bar{\mathbf{X}}_y - \bar{\mathbf{X}})(\bar{\mathbf{X}}_y - \bar{\mathbf{X}})^\top,$$

where $\bar{\mathbf{X}}_y$ the average predictor vector in slice y , $\bar{\mathbf{X}} = \sum_y (n_y/n) \bar{\mathbf{X}}_y$ is the overall average, and h is the number of slices.

2.3. Structured principal fitted components (SPFC)

This model was introduced by Cook and Forzani (2009b). The coordinate vectors ν_y are again modeled as in (2), but now a linear structure is used to model $\Delta = \sum_{i=1}^m \delta_i \mathbf{G}_i$, where $m \leq p(p+1)/2$, $\mathbf{G}_1, \dots, \mathbf{G}_m$ are known real symmetric $p \times p$ linearly independent matrices and the elements of $\delta = (\delta_1, \dots, \delta_m)^\top$ are functionally independent. It is required also that Δ^{-1} have the same linear structure as Δ : $\Delta^{-1} = \sum_{i=1}^m s_i \mathbf{G}_i$. To model a diagonal Δ we set $\mathbf{G}_i = \mathbf{e}_i \mathbf{e}_i^\top$, where $\mathbf{e}_i \in \mathbb{R}^p$ contains a 1 in the i -th position and zeros elsewhere. This basic structure can be modified straightforwardly to allow for a diagonal Δ with sets of equal diagonal elements, and for a non-diagonal Δ with equal off-diagonal entries and equal diagonal entries. In the latter case, there are only two matrices $\mathbf{G}_1 = \mathbf{I}_p$ and $\mathbf{G}_2 = \mathbf{e} \mathbf{e}^\top$, where $\mathbf{e} \in \mathbb{R}^p$ has all elements equal to 1.

In this case the central subspace $\mathcal{S}_{Y|\mathbf{X}} = \Delta^{-1} \mathcal{M}$ and then $\mu_y = \mu + \Delta \alpha \beta \mathbf{f}_y$. The MLE for $\mathcal{S}_{Y|\mathbf{X}}$ is the span of the first d eigenvectors of $\hat{\Delta}^{-1} \tilde{\Sigma}$ where $\hat{\Delta}$ indicates the MLE of Δ . The MLE $\hat{\Delta}$ evidently does not have a closed-form representation and an iterative algorithm is required for its computation; an algorithm was described in (Cook and Forzani 2009b).

2.4. Extended principal fitted components (EPFC)

In this version the coordinate vectors $\boldsymbol{\nu}_y$ are again modeled like (2) so that $\boldsymbol{\mu}_y = \boldsymbol{\mu} + \boldsymbol{\alpha}\boldsymbol{\beta}\mathbf{f}_y$, but now $\boldsymbol{\Delta}$ is modeled as $\boldsymbol{\Delta} = \boldsymbol{\alpha}\boldsymbol{\Omega}\boldsymbol{\alpha}^\top + \boldsymbol{\alpha}_0\boldsymbol{\Omega}_0\boldsymbol{\alpha}_0^\top$, where $\boldsymbol{\Omega} \in \mathbb{R}^{d \times d}$ and $\boldsymbol{\Omega}_0 \in \mathbb{R}^{p-d, p-d}$ are positive definite matrices, $(\boldsymbol{\alpha}, \boldsymbol{\alpha}_0)$ is an orthogonal matrix, and still $\mathcal{S}_{Y|\mathbf{X}} = \text{span}(\boldsymbol{\alpha})$. Then the MLE for $\mathcal{S}_{Y|\mathbf{X}}$ maximizes over $\mathcal{S}(\boldsymbol{\alpha}) \in \mathcal{G}_{(d,p)}$, the Grassmann manifold, the log likelihood function

$$L_d(\boldsymbol{\alpha}) = -\frac{np}{2}(1 + \log(2\pi)) - \frac{n}{2} \log |\tilde{\boldsymbol{\Sigma}}| - \frac{n}{2} \log |\boldsymbol{\alpha}^\top \tilde{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\alpha}| - \frac{n}{2} \log |\boldsymbol{\alpha}^\top \tilde{\boldsymbol{\Sigma}}_{\text{res}} \boldsymbol{\alpha}|,$$

where $\tilde{\boldsymbol{\Sigma}}_{\text{res}} = \tilde{\boldsymbol{\Sigma}} - \tilde{\boldsymbol{\Sigma}}_{\text{fit}}$ with $\tilde{\boldsymbol{\Sigma}}_{\text{fit}}$ defined in the IPFC model. See Cook (2007) for discussion.

2.5. Principal fitted components (PFC)

In this model, which was studied by Cook and Forzani (2009b) the coordinate vectors $\boldsymbol{\nu}_y$ are again modeled as in (2), but $\boldsymbol{\Delta}$ is now an unstructured positive definite matrix that is functionally independent of y . The central subspace is again $\mathcal{S}_{Y|\mathbf{X}} = \boldsymbol{\Delta}^{-1}\mathcal{M}$ and the MLE of $\mathcal{S}_{Y|\mathbf{X}}$ is the span of $\tilde{\boldsymbol{\Sigma}}_{\text{res}}^{-1/2} \mathbf{v}$, where the columns of $\mathbf{v} \in \mathbb{R}^{p \times d}$ are the first d eigenvectors of $\tilde{\boldsymbol{\Sigma}}_{\text{res}}^{-1/2} \tilde{\boldsymbol{\Sigma}} \tilde{\boldsymbol{\Sigma}}_{\text{res}}^{-1/2}$ and $\tilde{\boldsymbol{\Sigma}}_{\text{res}}$ was defined in the EPFC model.

2.6. Covariance reduction (CORE)

Covariance reduction addresses a rather different dimension reduction issue. Consider the problem of characterizing the covariance matrices $\boldsymbol{\Delta}_y$, $y = 1, \dots, h$, of a random vector \mathbf{X} observed in each of h normal populations. There is no interest in the population means $\boldsymbol{\mu}_y$. The methodology reviewed here, which was proposed by Cook and Forzani (2008), is based on reducing the sample covariance matrices to an informational core that is sufficient to characterize the variance heterogeneity across the h populations. It may be a useful alternative to the spectral modeling in many applications.

For this problem the central subspace, $\mathcal{S}_{Y|\mathbf{X}} = \text{span}(\boldsymbol{\alpha})$ is redefined to be the smallest subspace satisfying $\boldsymbol{\Delta}_y = \boldsymbol{\Delta} + P_{\boldsymbol{\alpha}(\boldsymbol{\Delta}_y)}^\top (\boldsymbol{\Delta}_y - \boldsymbol{\Delta}) P_{\boldsymbol{\alpha}(\boldsymbol{\Delta}_y)}$, where $P_{\boldsymbol{\alpha}(\boldsymbol{\Delta}_y)} = \boldsymbol{\alpha}(\boldsymbol{\alpha}^\top \boldsymbol{\Delta}_y \boldsymbol{\alpha})^{-1} \boldsymbol{\alpha}^\top \boldsymbol{\Delta}_y$ is the projection onto $\mathcal{S}_{Y|\mathbf{X}}$ using the inner product defined by $\boldsymbol{\Delta}_y$. We assume that the data consist of n_y independent observations of \mathbf{X}_y , $y = 1, \dots, h$. Then the MLE of $\mathcal{S}_{Y|\mathbf{X}}$ is the subspace that maximizes over $\mathcal{G}_{(d,p)}$ the log likelihood function

$$L_d(\boldsymbol{\alpha}) = c - \frac{n}{2} \log |\tilde{\boldsymbol{\Delta}}| + \frac{n}{2} \log |\boldsymbol{\alpha}^\top \tilde{\boldsymbol{\Delta}} \boldsymbol{\alpha}| - \sum_{y=1}^h \frac{n_y}{2} \log |\boldsymbol{\alpha}^\top \tilde{\boldsymbol{\Delta}}_y \boldsymbol{\alpha}|, \quad (3)$$

where c is a constant depending only on p and n_y , $\tilde{\boldsymbol{\Delta}}_y$, $y = 1, \dots, h$, denotes the sample covariance matrix from population y computed with divisor n_y , and $\tilde{\boldsymbol{\Delta}} = \sum_{y=1}^h (n_y/n) \tilde{\boldsymbol{\Delta}}_y$.

2.7. Likelihood acquired directions (LAD)

In this model, which was proposed by Cook and Forzani (2009a), we consider both a general conditional covariance $\boldsymbol{\Delta}_y$ and a general mean $\boldsymbol{\mu}_y$. It requires a categorical response Y . When the response is continuous or many-valued it is typical to follow (Li 1991) and replace it with a categorical version constructed by partitioning its range into h slices. The central

subspace $\mathcal{S}_{Y|\mathbf{X}} = \text{span}(\boldsymbol{\alpha})$ is the smallest subspace that satisfies the conditions (i) $\boldsymbol{\Delta}_y = \boldsymbol{\Delta} + P_{\boldsymbol{\alpha}(\boldsymbol{\Delta}_y)}^\top(\boldsymbol{\Delta}_y - \boldsymbol{\Delta})P_{\boldsymbol{\alpha}(\boldsymbol{\Delta}_y)}$ and (ii) $\text{span}(\boldsymbol{\alpha}) \subseteq \boldsymbol{\Delta}^{-1}\text{span}(\boldsymbol{\mu}_y - \boldsymbol{\mu})$. Condition (i) is the same as that encountered in the CORE model, while condition (ii) incorporates the conditional means $\boldsymbol{\mu}_y$. Assume as before that the data consist of n_y independent observations on \mathbf{X}_y , $y = 1, \dots, h$. Then the MLE for $\mathcal{S}_{Y|\mathbf{X}}$ maximizes over $\text{span}(\boldsymbol{\alpha}) \in \mathcal{G}_{(d,p)}$ the log likelihood function

$$L_d(\boldsymbol{\alpha}) = -\frac{np}{2}(1 + \log(2\pi)) - \frac{n}{2} \log |\tilde{\boldsymbol{\Sigma}}| + \frac{n}{2} \log |\boldsymbol{\alpha}\tilde{\boldsymbol{\Sigma}}\boldsymbol{\alpha}| - \frac{1}{2} \sum_{y=1}^h n_y \log |\boldsymbol{\alpha}\tilde{\boldsymbol{\Delta}}_y\boldsymbol{\alpha}| \quad (4)$$

where $\tilde{\boldsymbol{\Delta}}_y$ is as defined in the CORE model.

The likelihood functions $L_d(\boldsymbol{\alpha})$ presented so far have a similar feel and depend on similar quantities. They differ because of the structure placed on the conditional means $\boldsymbol{\mu}_y$ and conditional variances $\boldsymbol{\Delta}_y$, and because of the dimension reduction target. For example, the final terms in the likelihood functions for CORE (3) and LAD (4) are the same. The second and third terms are similar, with CORE using $\tilde{\boldsymbol{\Delta}}$ and LAD using $\tilde{\boldsymbol{\Sigma}}$. This difference arises because LAD extracts dimension reduction information from both the sample means $\bar{\mathbf{X}}_y$ and sample variances $\tilde{\boldsymbol{\Delta}}_y$, while CORE uses information from just the sample variances. The sample version $\tilde{\boldsymbol{\Sigma}} = \tilde{\boldsymbol{\Sigma}}_{\text{fit}} + \tilde{\boldsymbol{\Delta}}$ of the usual ANOVA identity $\text{VAR}(\mathbf{X}) = \text{VAR}(\text{E}(\mathbf{X}|Y)) + \text{E}(\text{VAR}(\mathbf{X}|Y))$ shows how information from the sample means $\bar{\mathbf{X}}_y$ enters $\tilde{\boldsymbol{\Sigma}}$ via $\tilde{\boldsymbol{\Sigma}}_{\text{fit}}$.

2.8. Envelope models for multivariate linear regression (MLM)

The dimension reduction methodology described in this section is different from the previous methods because it involves a multivariate response $\mathbf{Y} \in \mathbb{R}^r$ and a multivariate predictor $\mathbf{X} \in \mathbb{R}^p$, assumed to be non-stochastic, that are related through the standard normal multivariate linear model:

$$\mathbf{Y} = \boldsymbol{\beta}_0 + \boldsymbol{\beta}\mathbf{X} + \boldsymbol{\varepsilon}, \quad (5)$$

where $\boldsymbol{\beta}_0 \in \mathbb{R}^r$, $\boldsymbol{\beta} \in \mathbb{R}^{r \times p}$ and $\boldsymbol{\varepsilon} \sim N_r(0, \boldsymbol{\Sigma})$. The number of responses r is often large in modern applications of this model. In such situations the response vector may contain irrelevant linear combinations whose conditional distribution does not change with \mathbf{X} . It may also contain redundant information characterized by $\boldsymbol{\Sigma}$ having relatively large eigenvectors. Developed recently by [Cook, Li, and Chiaromonte \(2010\)](#), envelope models allow for the possibility that such irrelevant or redundant information may be present in \mathbf{Y} , and can yield an MLE of $\boldsymbol{\beta}$ with substantially smaller variation than the usual MLE.

The redundant and irrelevant information in (5) can be characterized by using the reducing subspaces of $\boldsymbol{\Sigma}$. Let $(\boldsymbol{\Gamma}, \boldsymbol{\Gamma}_0) \in \mathbb{R}^{r \times r}$ be an orthogonal matrix, where the columns of $\boldsymbol{\Gamma} \in \mathbb{R}^{r \times d}$ form a basis for the smallest reducing subspace of $\boldsymbol{\Sigma}$ that contains $\text{span}(\boldsymbol{\beta})$, which is called the $\boldsymbol{\Sigma}$ -envelope of $\text{span}(\boldsymbol{\beta})$. Then the parameters in model (5) have the structure $\boldsymbol{\beta} = \boldsymbol{\Gamma}\boldsymbol{\eta}$, and $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Omega}\boldsymbol{\Gamma}^\top + \boldsymbol{\Gamma}_0\boldsymbol{\Omega}_0\boldsymbol{\Gamma}_0^\top$, where $\boldsymbol{\eta} \in \mathbb{R}^{d \times r}$ is an unconstrained coordinate matrix, and $\boldsymbol{\Omega} \in \mathbb{R}^{d \times d}$ and $\boldsymbol{\Omega}_0 \in \mathbb{R}^{r-d \times r-d}$ are positive definite matrices. The theory developed by [Cook et al. \(2010\)](#) shows that substantial gains in efficiency are possible if the linear combinations $\boldsymbol{\Gamma}_0^\top \mathbf{Y}$ contain both the irrelevant and redundant information in \mathbf{Y} .

Fitting with this parameter structure requires estimation of $\boldsymbol{\beta}_0$, $\text{span}(\boldsymbol{\Gamma})$, $\boldsymbol{\eta}$, $\boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_0$. The partially maximized log likelihood for estimating $\text{span}(\boldsymbol{\Gamma})$ is similar to the objective function for extended principal fitted components described in Section 2.4.

3. Software overview

The software is organized to allow users to run different types of processing by calling just a few interface functions. Several auxiliary tools are then internally called to perform computations according to the options specified in those interface functions. As the main example, a single function called `ldr` manages all methods for maximum likelihood estimation of the central subspace. Arguments in this function specify the model to use, the response type – classified broadly into (a) continuous and (b) discrete or categorical – and the protocol for choosing the dimension d . This protocol allows d to be specified or estimated using an information criterion, a likelihood-ratio test or in some cases a permutation test. Separate functions `mlm` are used for the envelope methodology discussed in Section 2.8. In Section 4 we describe how to use the software in detail.

The package is designed to run mainly from the command-line, but we have also built a simple graphical user interface to make usage more intuitive and analysis more interactive. Although the graphical interface does not currently provide all the features available from the command-line based interface, it is powerful enough to run prototype computations using standard values for some parameters of the models. A brief tutorial for using this interface is given in Section 4.5.

Our implementation emphasizes easy-reading code over performance. It is mainly based upon built-in functions in MATLAB and has been tested for compatibility starting up from version 6.5 release 13. Nevertheless, some functions require the **Statistics Toolbox** as well. The **LDR** package relies on the `sg_min` (Lippert 2007) toolkit for gradient-based optimization over Stiefel-Grassmann manifolds (Lippert and Edelman 2000). Some minor modifications of `sg_min` were necessary for this application and this modified version of `sg_min` is available with our package. The modifications are described in the documentation that comes with the code.

3.1. Available methods

The **LDR** package offers an implementation of the models described in Section 2. SIR, SAVE and DR are incorporated as separate MATLAB functions with the same names. SIR and SAVE are available in Weisberg’s `dr` package, and DR is available from its authors (Li and Wang

Model	AIC	BIC	LRT	PERM
PC				
SIR				
SAVE				
DR				
IPFC	✓	✓	✓	
SPFC	✓	✓	✓	
EPFC	✓	✓	✓	
PFC	✓	✓	✓	
LAD	✓	✓	✓	✓
CORE	✓	✓	✓	✓

Table 1: Available models and testing criteria for d .

2007). We have included basic implementations here in order to provide a more self-contained environment for dimension reduction. We have also used a separate MATLAB function `PC` for the PC model, since this model gives the same solution as the usual principal components and we do not have a novel way to estimate d .

For the rest of the models, which are the primary focus of the **LDR** package, we have incorporated methods to estimate the central subspace, including estimation of its dimension d . Available procedures for choosing d include information criteria such as AIC and BIC (Burnham and Anderson 2002), likelihood-ratio testing, and permutation testing (Cook and Yin 2001). Table 1 shows the available testing methods for each model. All of these functions run both for continuous and categorical responses and share the common interface function `ldr`.

4. Using the software

4.1. Methods for estimating the central subspace

Except for envelope models, a single function called `ldr` provides a unified interface for all dimension reduction methods based on maximum likelihood estimation of the central subspace. Other methods such as SIR, SAVE, DR and PC are called from separate interface functions. Their usage is very similar and is discussed briefly in this section.

To start using the software, set the current directory to the package's location in your disk and type `setpaths`. This command adds all the directories in the package to the MATLAB path, so that all functions there become available for computation.

The data should be read and stored. From now on, let $Y \in \mathbb{R}^n$ denote a vector containing the response and let $X \in \mathbb{R}^{n \times p}$ denote the matrix of the predictors. Interface functions like `ldr` have workspace variables as arguments and do not allow managing data files directly. Despite several procedures for reading data being already available in MATLAB, they usually do not provide support for data files with headers. For that purpose, we have included functions `loadDATA` and `getDATA`. A detailed description of these commands is available through the software documentation.

Five arguments are mandatory when calling the `ldr` function and they must be provided in the order that follows: (i) the response vector Y ; (ii) the matrix of predictors X ; (iii) an identifier for the model to be used, such as 'PFC' or 'epfc'; (iv) the type of response, which may be 'cont' or 'disc' depending on whether Y is continuous (many-valued) or discrete which includes categorical; and (v) the dimension d of the central subspace or an identifier such as 'BIC' for an estimation method along with any necessary argument. Acronyms previously introduced in Section 2 have been used to identify the available models. Identifiers for estimation of d are 'AIC' for Akaike's information criterion, 'BIC' for Bayes information criterion, 'LRT' for likelihood-ratio test and 'PERM' for permutation test. Matching is case-insensitive, so you can type either 'LRT' or 'lrt'. The identifiers 'LRT' and 'PERM' should be followed by a test level, typically .01 or .05.

Outputs are the same for all models and they are: (i) the coordinates of the projection of the predictors onto the estimated central subspace, i.e. $\alpha^\top X$; (ii) a matrix whose columns are the estimated basis vectors for the central subspace; i.e. α ; (iii) the maximum value of the log likelihood; and the (iv) the estimated dimension in case a criterion for d was used.

Depending on the model, additional and optional arguments can be given. As an example, to slice a continuous response into ten slices for processing data under the LAD model, an additional string ‘`nslices`’ and the value 10 should be added to the arguments. In this case, a call to the function `ldr` should look like:

```
>> [WX,W,L,d] = ldr(Y, X, 'LAD', 'cont', 'aic', 'nslices', 10);
```

where `WX` is the projection of the predictors onto the estimated central subspace, `W` is a matrix whose columns are the estimated basis vectors for the central subspace, `L` is the maximum value of the log likelihood and `d` is the dimension of the central subspace estimated by using Akaike’s information criterion.

Methods such as EPFC, LAD and CORE require a starting basis for the central subspace prior to iteration. We determine the starting basis internally by searching over several estimators like SIR, SAVE, DR, PLS and PC and selecting the one that gives the largest value of the likelihood. Nevertheless, in some cases an external starting basis may be useful. A starting basis can be supplied by adding the text string ‘`initval`’ followed by a matrix whose column form a starting basis. Iterative estimation of the central subspace is carried out using the `sg_min` package. All the available parameters in this toolkit have been retained as optional inputs in LDR. In addition, the maximum number of iterations can be specified. By default, computation will stop after 1000 iterations if convergence has not been achieved. It should be noted that, since all the starting values we use are \sqrt{n} consistent estimators, the answer after one iteration is asymptotically as efficient as the MLE. The previous call to LAD with an initial basis $\mathbf{B} \in \mathbb{R}^{p \times d}$ for the central subspace and with no more than 100 iterations would look like

```
>> [WX,W,f,d] = ldr(Y, X, 'LAD', 'cont', 'aic', 'nslices', 10, ...
    'initval', B, 'maxiter', 100);
```

A list of the remaining optional arguments and their identifiers is given in Table 2. The optional commands may be given in any order.

4.2. Example with Y categorical

To further illustrate how to use the package, consider the problem of studying classification of hand-written digits. In this case the response is the digit $Y = \{0, 1, \dots, 9\}$, which is treated as categorical. The 44 subjects were asked to write 250 random digits. Each digit yields a 16-dimensional feature vector, consisting of 8 pairs of randomly sampled two-dimensional locations on the digit. The 44 subjects were divided into two groups of size 30 and 14, in which the first formed the training set with sample size 7,494 and the second formed the test set with sample size 3,498. The data set is available from the machine learning repository at the University of California at Irvine (<ftp://ftp.ics.uci.edu/pub/machine-learning-databases/pendigits>).

We focus on dimension reduction of the 16-dimensional feature vectors for the training set, which may serve as a preparatory step for developing an efficient classifier. For clarity, we consider only the digits 0, 6 and 9, so for this example $Y = \{0, 6, 9\}$. The reduced data set comprises 2,219 cases. The data are stored in the text file `digits.txt` where the first column takes values 1, 2 or 3 indicating if the observation corresponds to a 0, 6 or 9 respectively. We first load the response vector and the predictors by typing:

Model	Input	Description	Default value
SIR	'nslices'	Number of slices for continuous response preprocessing.	5 slices.
SAVE	'nslices'	Number of slices for continuous response preprocessing.	5 slices.
DR	'nslices'	Number of slices for continuous response preprocessing.	5 slices.
IPFC	'fy'	Basis for regression. A matrix must be given after this output.	Polynomial basis of degree $r = \max\{3, d + 1\}$.
	'alpha'	Confidence level for likelihood ratio test.	0.05
SPFC	'fy'	Basis for regression. A matrix must be given after this output.	Polynomial basis of degree $r = \max\{3, d + 1\}$.
	'alpha'	Confidence level for likelihood ratio test.	0.05
PFC	'fy'	Basis for regression. A matrix must be given after this output.	Polynomial basis of degree $r = \max\{3, d + 1\}$.
	'alpha'	Confidence level for likelihood ratio test.	0.05
EPFC	'fy'	Basis for regression. A matrix must be given after this output.	Polynomial basis of degree $r = \max\{3, d + 1\}$.
	'alpha'	Confidence level for likelihood ratio test.	0.05
	'initval'	Starting basis for central subspace estimation.	See text for details.
	'maxiter' see sg_min	Maximum number of iterations. Several optional inputs for optimization.	10000 'prcg', 'euclidean'
LAD	'nslices'	Number of slices for continuous response preprocessing.	5 slices.
	'alpha'	Confidence level for likelihood ratio test and permutation test.	0.05
	'npermute'	Permutations for permutation test.	500 permutations.
	'initval'	Starting basis for central subspace estimation.	See text for details.
	'maxiter' see sg_min	Maximum number of iterations. Several optional inputs for optimization.	10000 'prcg', 'euclidean'
CORE	'nslices'	Number of slices for continuous response preprocessing.	5 slices.
	'alpha'	Confidence level for likelihood ratio test and permutation test.	0.05
	'npermute'	Permutations for permutation test.	500 permutations.
	'initval'	Starting basis for central subspace estimation.	See text for details.
	'maxiter' see sg_min	Maximum number of iterations. Several optional inputs for optimization.	10000 'prcg', 'euclidean'

Table 2: Additional/optional inputs for available models.

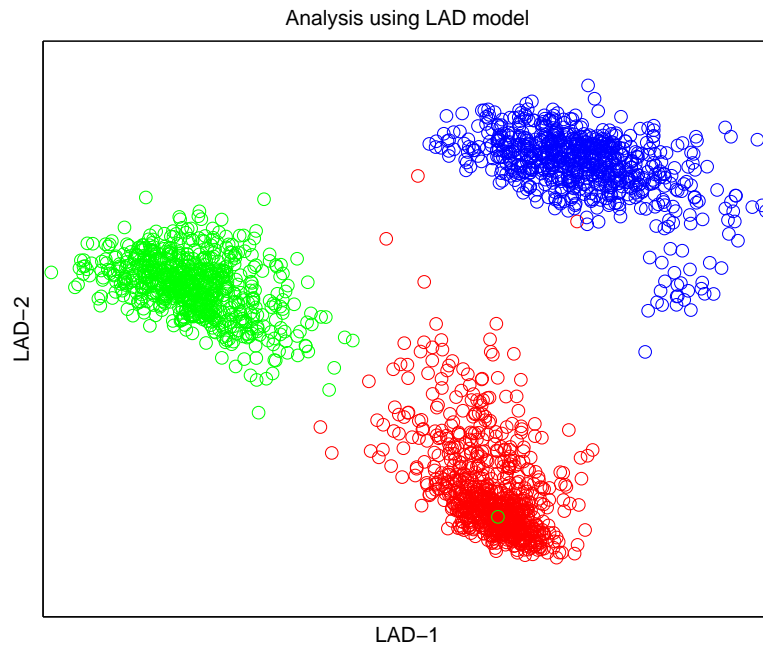


Figure 1: Plot of the two reduced LAD predictors with points marked to indicate the digit.

```
>> data = load('digits.txt');
>> Y = data(:, 1);
>> X = data(:, 2:end);
```

We next fit the LAD model $\mathbf{X}|(y = y) \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Delta}_y)$ looking for $d = 2$ linear combinations of the predictors:

```
>> [WX,W] = ldr(Y, X, 'LAD', 'disc', 2);
```

Here we used default options for optimization. The reduced predictors WX can be used later to get an efficient classifier. W is a basis for the estimated central subspace. To plot results, we can call the provided function `plotDR` as:

```
>> plotDR(WX, Y, 'disc', 'LAD');
```

This plots the two new reduced predictors WX using different colors to indicate if the response is 0, 6 or 9, as shown in Figure 1.

The function `plotDR` behaves like standard MATLAB plotting functions. For instance, a second call to `plotDR` will overwrite the first plot, unless the MATLAB command for a new figure has been issued. To carry out a similar analysis for variance differences only we should use the CORE model and replace 'LAD' with 'CORE' in the statements above. To apply CORE with $d = 1$, the statement should be

```
>> [WX,W,f,d] = ldr(Y, X, 'CORE', 'disc', 1)
```

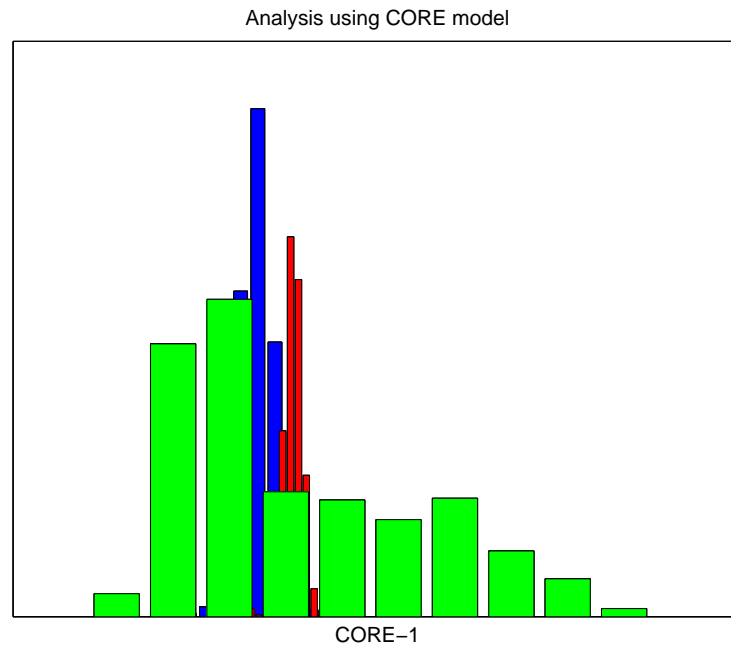


Figure 2: Histograms of the reduced CORE predictor by digit.

and the corresponding plot

```
>> plotDR(WX, Y, 'disc', 'CORE');
```

is the histogram of the predictors by digit shown in Figure 2.

If we allow the means μ_y to be different but constrain the covariance matrices Δ_y to be the same, then we are under the PFC model. Suppose we want to estimate d assuming the PFC model. We should then include a command like 'lrt', 'AIC', 'BIC' or 'perm' in the MATLAB statement:

```
>> [WX,W,f,d] = ldr(Y, X, 'PFC', 'disc', 'AIC')
```

will give $d = 2$, the number of dimensions that PFC indicates are necessary for reduction without loss of information using the AIC criterion. The plotting command is analogous to the other cases.

Under the same PFC model we could test to see if the covariance matrix of $\mathbf{X}|Y$ is diagonal. In such a case using diagonal structure could give a more efficient estimator. The lrt can be obtained from the command

```
>> testdiag4pfc(Y, X, 'disc', .05);
```

The answer returned for these data is 0, meaning that the null hypothesis is clearly rejected. Nevertheless, for illustration the command to fit PFC with a diagonal covariance matrix is

```
>> [WX,W,f,d] = ldr(Y, X, 'SPFC', 'disc', 'BIC');
```

where we are using BIC to find the dimension of the central subspace.

The EPFC model is also a particular case of PFC. But here it is assumed that the central subspace is an invariant subspace of the covariance matrix (see Section 2.4). The MATLAB command is the same but we should change ‘SPFC’ to ‘EPFC’.

When using SIR, SAVE, DR or PC the dimension d should be given, as no estimation method for d is currently available in the **LDR** package. If we choose $d = 2$, corresponding commands are

```
>> [WX,W] = SIR(Y, X, 'disc', 2);
>> [WX,W] = SAVE(Y, X, 'disc', 2);
>> [WX,W] = DR(Y, X, 'disc', 2);
```

The plot command `plotDR` is the same as before.

4.3. Example with Y continuous

Processing data with continuous responses is analogous, but it often allows for additional arguments. For illustration, consider the Naphthalene data. These data consist of 80 observations from an experiment to study the effects of three process variables in the vapor phase oxidation of naphthalene (Franklin, Pinchbeck, and Popper 1956). The data are stored in `nap.txt`. The response is the percentage conversion of naphthalene to naphthoquinone, and the three predictors are air to naphthalene ratio, contact time and bed temperature. Although there are only three predictors in this regression, dimension reduction may still be useful for visualization (Cook 1998).

Suppose you have already loaded your continuous data into workspace variables `Y` and `X` as (the complete script and data are available along with the paper and within the **LDR** package in the directory `papers/JSS_example`):

```
>> data = load('nap.txt');
>> X = data(:, [4 5 9]);
>> Y = data(:, 12);
```

The models LAD and CORE treat the response as categorical and therefore require the continuous response Y to be sliced (if not provided, 5 is the default number of slices.) To estimate under the LAD model using 6 slices and $d = 1$ we would use a statement such as

```
>> [WX,W,f,d] = ldr(Y, X, 'LAD', 'cont', 1, 'nslices', 10);
```

To plot the response Y versus the new predictor `WX` we use

```
>> plotDR(Y, WX, 'cont', 'LAD');
```

The result is shown in Figure 3. The dimension d could be estimated via likelihood ratio testing with level 5% by using the command

```
>> [WX,W,f,d] = ldr(Y, X, 'LAD', 'cont', 'lrt', 'nslices', 5, 'alpha', 0.05);
```

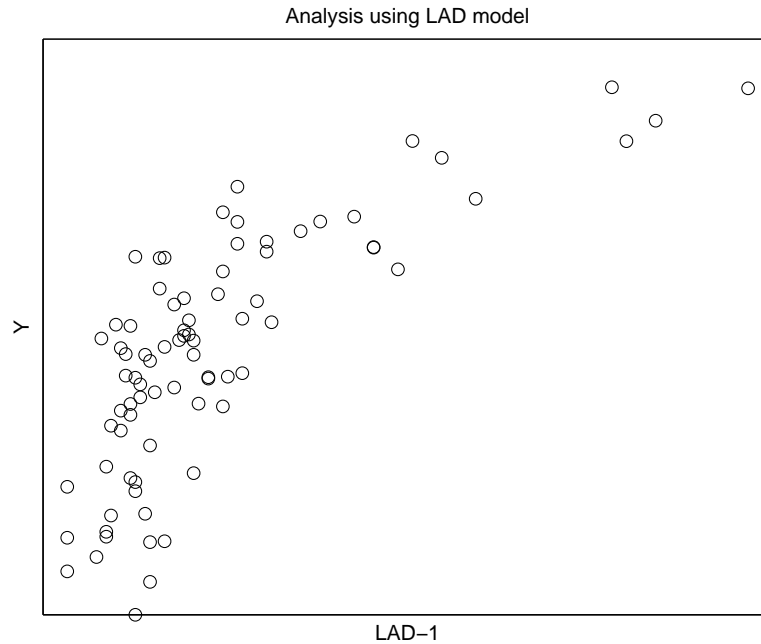


Figure 3: Plot of the continuous response Y vs the reduced predictor using LAD method for the nap data.

Similarly, to estimate d with 5% permutation test using 500 permutations use the command

```
>> [WX,W,f,d] = ldr(Y, X, 'LAD', 'cont', 'perm', 'nslices', 5, ...
    'alpha', 0.05, 'npermute', 500);
```

Permutation tests can take a while to run.

These arguments are inappropriate for models like IPFC, SPFC, EPFC and PFC since they do not use slicing but instead deal directly with Y as continuous. Therefore a basis \mathbf{f}_y for the regression of \mathbf{X} on Y should be supplied as an $n \times r$ matrix, say \mathbf{FY} , with rows \mathbf{f}_y^\top . Here n denotes the sample size and r denotes the dimension of the basis to be considered. In this package we provide an auxiliary function `get_fy` that returns \mathbf{f}_y as a polynomial of order r , for a given r . A call to the PFC model, for example, should look like:

```
>> FY = get_fy(Y, 2);
>> [WX,W,f,d] = ldr(Y, X, 'PFC', 'cont', 'bic', 'fy', FY);
```

where we have told the software to use a second-order polynomial for the regression of \mathbf{X} on Y and we are also estimating d using the Bayes information criterion. The dimension chosen under this method is $d = 1$. The result \mathbf{WX} of dimension 1 can be consider as the new predictor and the command

```
>> plotDR(Y, WX, 'cont', 'PFC');
```

will give the plot of the response Y versus the new predictor WX .

Further examples of how to use the software are provided in the `papers` folder within **LDR**. They reproduce results and figures published when introducing the methods discussed in Section 2 and can be used to explore the main features in the package.

4.4. Envelope models

The basic command for computing an estimated basis G of $\text{span}(\Gamma)$ in the envelope model (2.8) is `[GX, G, L, dhat] = mlm_fit(Y, X, dim)`, where `dim` stands for the dimension of the envelope. The essential output consists of the estimate G of Γ , the value L of the log likelihood at the MLE and the estimated dimension `dhat` of the envelope. `GX` is an exploratory quantity and not used routinely in this application. The following options are available for `dim`: If `dim` is set to an integer d , eg., `mlm_fit(Y, X, 2)`, then `dhat` = d and the envelope model is fitted with the specified dimension. If `dim` is set to 'aic' or 'bic' then the indicated information criterion is used to estimate d . These commands may take a long time, depending on the size of the regression. If `dim` is set to the pair 'lrt', .05, eg. `mlm_fit(Y, X, 'lrt', .05)`, then likelihood ratio testing at level .05 is used to estimate d .

The following commands are available following the initial fit that produces G .

- `[betaem, eta, Omega, Omega0, S1, S2] = mlm_empars(Y, X, G)` returns the estimated parameters from the fit of the envelope model. `betaem` = G x `eta` is the estimated coefficient matrix Ω and `Omega0` are the estimates of Ω and Ω_0 , `S1` is the estimate of $\Gamma\Omega\Gamma^\top$ and `S2` is the estimate of $\Gamma_0\Omega_0\Gamma_0^\top$.
- `mlm_emses(Y, X, G)` returns the matrix of standard errors of the elements `betaem` from the fit of the envelope model.
- `mlm_seratios(Y, X, G)` returns the matrix of ratios of standard errors of the full model and envelope model estimates of β . The ratios are (standard error for the full model) / (standard error for the envelope model).

Other commands are documented in the folder `mlm` that comes with the MATLAB code.

4.5. Using the graphical user interface

The graphical user interface (GUI) is expected to provide an easy-to-use tool to perform small prototype tasks using the functions available in the package. Figure 4 shows the layout of the interface with its main controls. The GUI is started by typing `demo` in the command window, provided the current directory is the **LDR** folder. Once it has been loaded, data analysis using the GUI typically follows a sequence of steps:

1. Load data by clicking on the `LOAD DATA` (1) button and then follow the dialog box to find the desired data file.
2. Set the response variable and the predictors by clicking on the `set Y` and `set X` buttons (2).
3. Specify the type of the response according to the data by choosing the right option in the popup menu (4). For continuous responses, you should also give the number of slices

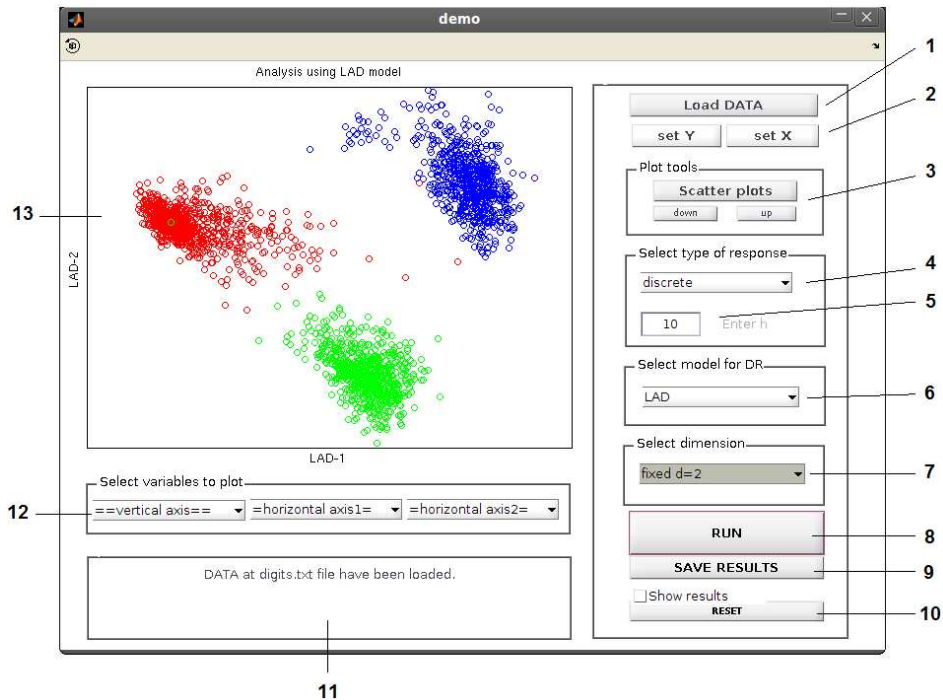


Figure 4: Lay out of the graphical user interface for the **LDR** package. (1) Press to open a dialog-box to search for a data file; (2) press to set the response and the predictors for the analysis; (3) press to sequentially display scatter plots for a given variable, set in the axes popups; (4) specify if the response is continuous or discrete; (5) edit text to pass basic parameters for processing; (6) choose the method to be applied; (7) choose the dimension of the subspace to look for, or a method to infer it; (8) press to select predictors and apply the dimension reduction method; (9) press to open a dialog-box to save the obtained results; (10) press to clear stored results; (11) look at brief comments and warnings; (12) choose the variable to plot in each axis; (13) plot of results.

for the slicing procedure in the editable text box (5) if you plan to apply methods such as LAD, CORE, SIR, SAVE or DR.

4. Select the dimension of the reduced subspace or choose an estimation method from the popup (7). Currently, the GUI allows only $d = 1, 2, 3$. The available estimation methods include all those listed in Table 1. Some estimation methods, in particular permutation tests, take a long computation time and can freeze the GUI for awhile.
5. Start the computation by clicking the RUN button (8).

By default, processing data with the GUI returns only results through plots. Nevertheless, you can check the **Show results** option to allow the program to print results on the command window and you can save results as text files by clicking on the **SAVE RESULTS** button (9). When you do so, you are first asked to select what results you want to save. All analysis performed so far are listed and you can choose as many as you like. For each selection, you are allowed to save two files. The first one contains the response plus the transformed

predictors, while the second one contains only the generating vectors for the reduced subspace. You can skip saving either of them by clicking on the **Cancel** button in the respective save dialog.

After the data are loaded, the popups (12) below the figure axes get filled with identifiers for each column in the data file. You can get scatter plots of pairs of variables by selecting one of them in the popup corresponding to the vertical axis and the other one in the popup menu related to the horizontal axis. For a sequential display of scatter plots for some given variable, you can instead set the related identifier in one of these popups and then click on the **Scatter plots** button (3). You can then move through the scatter plots by clicking the **up** and **down** buttons placed below the **Scatter plots** button.

5. Conclusion

This paper introduces new software for sufficient dimension reduction based on maximum likelihood estimation of the central subspace. MATLAB implementation of these methods allows easy integration of complex optimization tools and provides a flexible environment for graphics capabilities and further model development. Scripts are given to reproduce all published results concerning the methods discussed above, in agreement with the reproducible research philosophy. We plan to continue adding methods to the package and to finish code verification for full compatibility with the OCTAVE programming language in order to provide tools suitable to run under GNU software.

Acknowledgments

We are indebted to Eduardo Tabacman for his patient testing of the code and for bringing us some tips about the **sg_min** package. We also thank an anonymous reviewer for his suggestions to improve the code, and Ross Lippert for allowing us to distribute a modified version of his code with our package.

This work was supported in part by CONICET through grant PIP 5811, ANPCyT through grant PICT 2008-0622, Universidad Nacional del Litoral through grant CAI+D PI-12/M309 and National Science Foundation through grant DMS-1007547 awarded to RDC.

References

- Bura E, Cook RD (2001). “Estimating the Structural Dimension of Regressions via Parametric Inverse Regression.” *Journal of the Royal Statistical Society B*, **63**, 393–410.
- Bura E, Pfeiffer RM (2003). “Graphical Methods for Class Prediction Using Dimension Reduction Techniques on DNA Microarray Data.” *Bioinformatics*, **19**, 1252–1258.
- Burnham K, Anderson D (2002). *Model Selection and Multimodel Inference*. John Wiley & Sons, New York.
- Cook RD (1994). “Using Dimension Reduction Subspaces to Identify Important Inputs in Models of Physical Systems.” *Proceedings of the Section on Physical and Engineering Sciences, American Statistical Association*, pp. 18–25.

- Cook RD (1998). *Regression Graphics*. John Wiley & Sons, New York.
- Cook RD (2003). “Dimension Reduction and Graphical Exploration in Regression.” *Statistics in Medicine*, **22**, 1399–1413.
- Cook RD (2007). “Fisher Lecture: Dimension Reduction in Regression.” *Statistical Science*, **22**, 1–26.
- Cook RD, Forzani L (2008). “Covariance Reducing Models: An Alternative to Spectral Modeling of Covariance Matrices.” *Biometrika*, **95**(4), 799–812.
- Cook RD, Forzani L (2009a). “Likelihood-Based Sufficient Dimension Reduction.” *Journal of the American Statistical Association*, **104**(485), 197–208.
- Cook RD, Forzani L (2009b). “Principal Fitted Components in Regression.” *Statistical Science*, **23**, 485–501.
- Cook RD, Li B (2002). “Dimension Reduction for the Conditional Mean in Regression.” *The Annals of Statistics*, **30**, 455–474.
- Cook RD, Li B, Chiaromonte F (2010). “Envelope Models for Parsimonious and Efficient Multivariate Linear Regression.” *Statistica Sinica*, **20**, 927–1010.
- Cook RD, Ni B (2005). “Sufficient Dimension Reduction via Inverse Regression: A Minimum Discrepancy Approach.” *Journal of the American Statistical Association*, **100**, 410–428.
- Cook RD, Weisberg S (1991). “Discussion of Sliced Inverse Regression by K. C. Li.” *Journal of the American Statistical Association*, **86**, 328–332.
- Cook RD, Yin X (2001). “Dimension Reduction and Visualization in Discriminant Analysis.” *Australia New Zeland Journal of Statistics*, **43**, 147–199.
- Edelman A, Arias T, Smith ST (1999). “The Geometry of Algorithms with Orthogonality Constraints.” *SIAM Journal of Matrix Analysis and Applications*, **20**, 303–353.
- Franklin NL, Pinchbeck PH, Popper F (1956). “A Statistical Approach to Catalyst Development, Part 1: The Effects of Process Variables in the Vapor Phase Oxidation of Naphthalene.” *Transactions of the Institute of Chemical Engineers*, **34**, 867–889.
- Helland I (1990). “Partial Least Squares Regression and Statistical Models.” *Scandinavian Journal of Statistics*, **17**, 97–114.
- Huber PJ (1985). “Projection Pursuit.” *The Annals of Statistics*, **13**, 435–525.
- Li B, Wang S (2007). “On Directional Regression for Dimension Reduction.” *Journal of the American Statistical Association*, **102**, 997–1008.
- Li B, Zha H, Chiaromonte F (2005). “Contour Regression: A General Approach to Dimension Reduction.” *The Annals of Statistics*, **33**, 1580–1616.
- Li KC (1991). “Sliced Inverse Regression for Dimension Reduction.” *Journal of the American Statistical Association*, **86**, 316–342.

- Li KC (1992). “On Principal Hessian Directions for Data Visualization and Dimension Reduction: Another Application of Stein’s Lemma.” *Journal of the American Statistical Association*, **87**, 1025–1039.
- Li L, Li H (2004). “Dimension Reduction Methods for Microarrays with Application to Censored Survival Data.” *Bioinformatics*, **20**, 3406–3412.
- Lippert R (2007). “**sg_min**: Stiefel Grassmann Optimization.” MATLAB package version 2.4.1, URL <http://www-math.mit.edu/~lippert/sgmin.html>.
- Lippert R, Edelman A (2000). “Nonlinear Eigenvalue Problems with Orthogonality Constraints.” In Z Bai, J Demmel, J Dongarra, A Ruhe, H van der Vorst (eds.), *Templates for the Solution of Algebraic Eigenvalue Problems*. SIAM, Philadelphia.
- Pardoe I, Yin X, Cook RD (2007). “Graphical Tools for Quadratic Discriminant Analysis.” *Technometrics*, **49**, 172–183.
- Shao Y, Cook RD, Weisberg S (2007). “Marginal Tests with Sliced Average Variance Estimation.” *Biometrika*, **94**, 285–296.
- The MathWorks, Inc (2009). “MATLAB – The Language of Technical Computing.” URL <http://www.mathworks.com/products/matlab/>.
- Tibshirani R (1996). “Regression Shrinkage and Selection via the LASSO.” *Journal of the Royal Statistical Society B*, **58**, 267–288.
- Weisberg S (2002). “Dimension Reduction Regression in R.” *Journal of Statistical Software*, **7**(1), 1–22. URL <http://www.jstatsoft.org/v07/i01/>.
- Weisberg S (2009). “**dr**: Methods for Dimension Reduction in Regression.” R package version 3.0.4, URL <http://CRAN.R-project.org/package=dr>.
- Wold H (1996). “Soft Modeling by Latent Variables: the Nonlinear Iterative Partial Least Squares (NIPALS) Approach.” In *Perspectives in Probability and Statistics, In Honor of M. S. Bartlett*, pp. 117–144.
- Xia Y, Tong H, Li W, Zhu L (2002). “An Adaptive Estimation of Dimension Reduction Space.” *Journal of the Royal Statistical Society B*, **64**, 363–410.
- Ye Z, Weiss R (2003). “Using the Bootstrap to Select One of a New Class of Dimension Reduction Methods.” *Journal of the American Statistical Association*, **98**, 968–978.
- Zhu L, Ohtaki M, Li Y (2005). “On Hybrid Methods of Inverse Regression-Based Algorithms.” *Computational Statistics and Data Analysis*, **51**, 2621–2635.
- Zhu Y, Zheng P (2006). “Fourier Methods for Estimating the Central Subspace and the Mean Subspace in Regression.” *Journal of the American Statistical Association*, **101**, 1638–1651.

Affiliation:

R. Dennis Cook
School of Statistics
University of Minnesota
E-mail: dennis@stat.umn.edu
URL: <http://www.stat.umn.edu/~dennis/>

Liliana M. Forzani
IMAL and Department of Mathematics
FIQ-Universidad Nacional del Litoral – CONICET
3000 Santa Fe, Argentina
E-mail: liliana.forzani@gmail.com
URL: <http://liliana.forzani.googlepages.com/english>

Diego R. Tomassi
IMAL and Department of Informatics
FICH-Universidad Nacional del Litoral - CONICET
3000 Santa Fe, Argentina
E-mail: diegotomassi@gmail.com