High-dimensional Regression

Jan de Leeuw

Volume 2, pp. 816–818

in

Encyclopedia of Statistics in Behavioral Science

ISBN-13: 978-0-470-86080-9
ISBN-10: 0-470-86080-4

Editors

Brian S. Everitt & David C. Howell

High-dimensional Regression

In regression analysis, there are n observations y_i on a dependent variable (also known as outcome or criterion) that are related to n corresponding observations x_i on p independent variables (also known as inputs or predictors). Fitting regression models of some form or another is by far the most common uses of statistics in the sciences (*see* **Multiple Linear Regression**).

Statistical theory tells us to assume that the observed outcomes $\underline{y}_i$ are realizations of n random variables $\underline{y}_i$. We model the conditional expectation of $\underline{y}_i$ given x_i, or, to put it differently, we model the expected value of $\underline{y}_i$ as a function of x_i

$$\mathbf{E}(\underline{y}_i \mid x_i) = F(x_i), \tag{1}$$

where the function F must be estimated from the data. Often the function F is known except for a small number of parameters. This defines parametric regression. Sometimes F is unknown, except for the fact that we know that has a certain degree of continuity or smoothness. This defines **nonparametric regression**.

In this entry, we are specifically concerned with the situation in which the number of predictors is large. Through the years, the meaning of ‘large’ has changed. In the early 1900s, three was a large number, in the 1980s 100 was large, and at the moment we sometimes have to deal with situations in which there are 10 000 predictors. This means, in the regression context, that we have to estimate a function F of 10 000 variables. Modern data collection techniques in, for example, genetics, environmental monitoring, and consumer research lead to these huge datasets, and it is becoming clear that classical statistical techniques are useless for such data. Entirely different methods, sometimes discussed under the labels of ‘data mining’ or ‘machine learning’, are needed [5] (*see* **Data Mining**).

Until recently **multiple linear regression**, in which F is linear, was the only practical alternative to deal with a large number of predictors. Thus, we specialize our model to

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{s=1}^{p} \beta_s x_{is}. \tag{2}$$

It became clear rather soon that linear regression with a large number of predictors has many problems. The main ones are multicollinearity, often even singularity, and the resulting numerical instability of the estimated regression coefficients (*see* **Collinearity**).

An early attempt to improve this situation is using variable selection. We fit the model

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{s=1}^{p} \beta_s \delta_s x_{is}. \tag{3}$$

where δ_s is either zero or one. In fitting this model, we select a subset of the variables and then do a linear regression. Although variable selection methods appeared relatively early in the standard statistical packages, and became quite popular, they have the major handicap that they must solve the combinatorial problem of finding the optimum selection from among the 2^p possible ones. Since this rapidly becomes unsolvable in any reasonable amount of time, various heuristics have been devised. Because of the instability of high-dimensional linear regression problems, the various heuristics often lead to very different solutions. Two ways out of the dilemma, which both stay quite close to linear regression, have been proposed around 1980. The first is principal component regression (*see* **Principal Component Analysis**) or PCR, in which we have

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{t=1}^{q} \beta_t \left[\sum_{s=1}^{p} \alpha_{ts} x_{is}\right]. \tag{4}$$

Here we replace the p predictors by $q < p$ principal components and then perform the linear regression. This tackles the multicollinearity problem directly, but it inherits some of the problems of principal component analysis. How many components do we keep? And how do we scale our variables for the component analysis?

The second, somewhat more radical, solution is to use the **generalized additive model** or GAM discussed by [6]. This means

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{s=1}^{p} \beta_s \phi_s(x_{is}), \tag{5}$$

where we optimize the regression fit over both θ and the functions (transformations) ϕ. Usually we require $\phi \in \Phi$ where Φ is some finite dimensional subspace of functions, such as polynomials or splines with a given knot sequence. Best fits for such models are easily computed these days by using alternating least squares algorithms that iteratively alternate fitting θ for fixed ϕ and fitting ϕ for fixed θ [1]. Although generalized additive models add a great deal of flexibility to the regression situation, they do not directly deal with the instability and multicollinearity that comes from the very large number of predictors. They do not address the data reduction problem, they just add more parameters to obtain a better fit.

A next step is to combine the ideas of PCR and GAM into projection pursuit regression or PPR [4]. The model now is

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{t=1}^{q} \phi_t \left[\sum_{s=1}^{p} \alpha_{ts} x_{is} \right]. \tag{6}$$

This is very much like GAM, but the transformations are applied to a presumably small number of linear combinations of the original variables. PPR regression models are closely related to neural networks, in which the linear combinations are the single hidden layer and the nonlinear transformations are sigmoids or other squashers (*see* **Neural Networks**). PPR models can be fit by general neural network algorithms.

PPR regression is generalized in Li's **slicing inverse regression** or SIR [7, 8], in which the model is

$$\mathbf{E}(\underline{y}_i \mid x_i) = F \left[\sum_{s=1}^{p} \alpha_{1s} x_{is}, \ldots, \sum_{s=1}^{p} \alpha_{qs} x_{is} \right]. \tag{7}$$

For details on the SIR and PHD algorithms, we refer to (*see* **Slicing Inverse Regression**).

Another common, and very general approach, is to use a finite basis of functions h_{st}, with $t = 1, \ldots, q_s$, for each of the predictors x_s. The basis functions can be polynomials, piecewise polynomials, or splines, or radical basis functions. We then approximate the multivariate function F by a sum of products of these basis functions. Thus we obtain the model

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{t_1=1}^{q_1} \cdots \sum_{t_p=1}^{q_p} \theta_{t_1 \cdots t_p} \times h_{1t_1}(x_{i1}) \times \cdots \times h_{pt_p}(x_{ip}) \tag{8}$$

This approach is used in multivariate adaptive regression splines, or MARS, by [3]. The basis functions are splines, and they adapt to the data by locating the knots of the splines.

A different strategy is to use the fact that any multivariate function can be approximated by a multivariate step function. This fits into the product model, if we realize that multivariate functions constant on rectangles are products of univariate functions constant on intervals. In general, we fit

$$\mathbf{E}(\underline{y}_i \mid x_i) = \sum_{t=1}^{q} \theta_t I(x_i \in R_t). \tag{9}$$

Here, the R_t define a partitioning of the p-dimensional space of predictors, and the $I()$ are indicator functions of the q regions. In each of the regions the regression function is a constant. The problem, of course, is how to define the regions. The most popular solution is to use a recursive partitioning algorithm such as **Classification and Regression Trees**, or by the algorithm CART [2], which defines the regions as rectangles in variable space. Partitionings are refined by splitting along a variable, and by finding the variable and the split which minimize the residual sum of squares. If the variable is categorical, we split into two arbitrary subsets of categories. If the variable is quantitative, we split an interval into two pieces. This recursive partitioning builds up a binary tree, in which leaves are refined in each stage by splitting the rectangles into two parts.

It is difficult, at the moment, to suggest a best technique for high-dimensional regression. Formal statistical **sensitivity analysis**, in the form of standard errors and confidence intervals, is largely missing. Decision procedures, in the form of tests, are also in their infancy. The emphasis is on exploration and on computation. Since the data sets are often enormous, we do not really have to worry too much about significance, we just have to worry about predictive performance and about finding (mining) interesting aspects of the data.

References

[1] Breiman, L. & Friedman, J.H. (1985). Estimating optimal transformations for multiple regression and correlation, *Journal of the American Statistical Association* **80**, 580–619.

[2] Breiman, L., Friedman, J., Olshen, R. & Stone, C. (1984). *Classification and Regression Trees*, Wadsworth.

[3] Friedman, J. (1991). Multivariate adaptive regression splines (with discussion), *Annals of Statistics* **19**, 1–141.

[4] Friedman, J. & Stuetzle, W. (1981). Projection pursuit regression, *Journal of the American Statistical Association* **76**, 817–823.

[5] Hastie, T., Tibshirani, R. & Friedman, J. (2001). *The Elements of Statistical Learning*, Springer.

[6] Hastie, T.J. & Tibshirani, R.J. (1990). *Generalized Additive Models*, Chapman and Hall, London.

[7] Li, K.C. (1991). Sliced inverse regression for dimension reduction (with discussion), *Journal of the American Statistical Association* **86**, 316–342.

[8] Li, K.C. (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.

JAN DE LEEUW