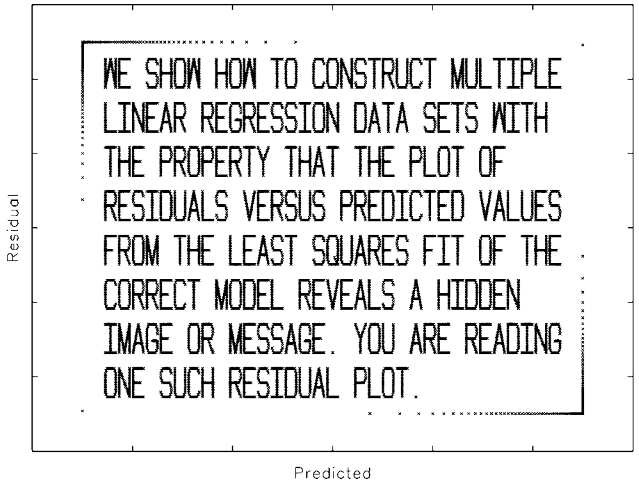
Statistical Computing and Graphics

Residual (Sur)Realism

Leonard A. STEFANSKI



KEY WORDS: Added-variable plot; Backward selection; Forward selection; Hidden image; Hidden message; Linear regression; Model selection; Partial regression plot; Residual plots; Variable selection.

PROLOGUE: REMARKABLE RESIDUALS

In a scene early in *Indiana Jones and the Last Crusade*, Professor Jones concludes his lecture on the archaeologist's endless search for lost antiquities with the admonition "… and **X** never, ever marks the spot." This scene portends a later one wherein Indy discovers a secret passageway to the Knight's Tomb under a giant Roman numeral ten embedded in the tiled floor of an old church, inducing him to chuckle "… ten, **X** marks the spot."

Statistics will never have the allure or the cachet of archaeology, and no professor of statistics will ever possess the Hollywood-imbued charm or magnetism of Professor Jones. Nevertheless, students of statistics can still experience the same sense of irony, humor, and interest-piquing discovery embodied in these scenes from *The Last Crusade*.

For example, imagine the reaction of a student who, upon completing an assignment of fitting a given multiple linear regression model and examining residual plots, is confronted with the residual plot in Figure 1(a), which contradicts G.E.P. Box's famous quotation about all models being wrong in the same way that Professor Jones's discovery under the Roman numeral ten contradicted his assertion that **X** never marks the spot (a residual plot version of which appears in Figure 1(b)). Of course, if the regression assignment is due just prior to a "big game," then the student might be more intrigued by residual plots of the sort in Figures 1(c) and (d). Figure 1(e) depicts Homer Simpson explaining how to embed images in regression residual plots. And if the residual plots in (a)–(e) are not attention-getting enough, the student who is unexpectedly confronted with the residual plot in Figure 1(f) is certainly going to be buffaloed (perhaps "bisoned" is taxonomically more correct but not grammatically).

1. INTRODUCTION

Several colleagues (D. Boos, S. K. Ghosh, H. Zhang, H. Bondell and L. Li) and I recently formed a research group to address problems of mutual interest in variable and model selection. Among the tasks that we set for ourselves were the construction of test datasets for illustrating and comparing the performances of the sundry approaches to variable selection available in the literature, and the construction of novel datasets for use in teaching variable selection in undergraduate and graduate level courses. This article is directed at the latter objective.

We show how to generate a linear regression dataset $\{X_{1,i}, \ldots, X_{p,i}, Y_i\}_{i=1}^n$ with the property that if the "correct" model is fit to the data, then the usual plot of residuals versus predicted values manifests a predetermined black-and-white image, that is, black pixels are mapped at the plotted (predicted, residual) pairs. The residual plots in Figure 1 are from six different, four-variable, multiple linear regression model datasets constructed using the algorithm in Section 2.

Apart from the obvious entertainment value, the primary pedagogical value of datasets so constructed lies in their ability to generate a certain "How-did-you-do-that?" curiosity among students that is not so easy to generate in other ways. The answer to this natural question is accessible to any student with a working knowledge of projection matrices at the level of Christensen (2002). Thus, graduate students and some advanced undergraduates should be capable of not only understanding the method of construction, but of programming it, and modifying the basic algorithm.

Leonard A. Stefanski is Professor, Department of Statistics, North Carolina State University, Raleigh, NC 27695-8203 (E-mail: *stefanski@stat.ncsu.edu*). The author acknowledges technical assistance from Homer Simpson, please see Figure 1(e), and financial support from the National Science Foundation, please see Figure 3. Also sincere thanks to the editorial staff for prompt handling of this article.

©2007 *American Statistical Association* *DOI: 10.1198/000313007X190079*

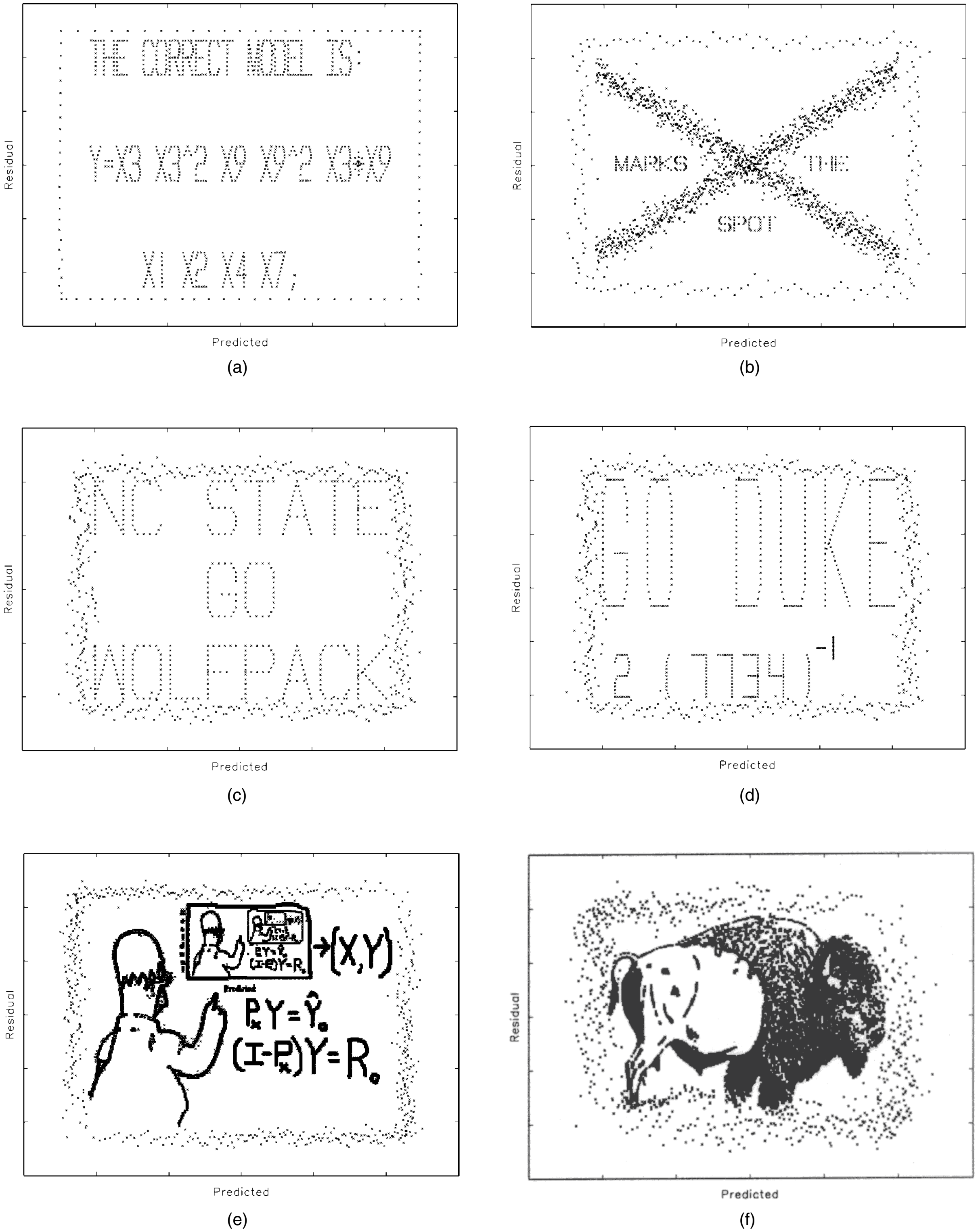


Figure 1. Some remarkable residual plots. Linear regression residuals are plotted versus predicted values for six different datasets.

A black-and-white digital image is essentially nothing more than a scatterplot of abscissa-ordinate pairs indicating the locations of the black pixels. Now consider labeling the abscissa "Predicted Values" and the ordinate "Residuals" and you have a residual plot that displays an image. Well, it is not quite that easy. Residuals and fitted values have certain mathematical properties that the ordinates and abscissae of an image generally do not possess, and this presents one hurdle to clear. However, the greater problem is working backwards from given target vectors of residuals and fitted values to find a dataset such that a regression model fit to the data has the desired target residuals and fitted values.

2. HOW DID YOU DO THAT?

Suppose that $\widehat{\mathbf{Y}}_0$ and $\mathbf{R}_0$ are given $n \times 1$ vectors of predicted values and residuals. Our interest lies in the case in which the pairs $(\widehat{\mathbf{Y}}_{0,i}, \mathbf{R}_{0,i})$, $i = 1, \ldots, n$, correspond to the black-pixel locations in a black-white image, and thus the image is manifest in the scatterplot of $\mathbf{R}_0$ versus $\widehat{\mathbf{Y}}_0$. We now show how to generate regression data, that is, an $n \times p$ matrix $\mathbf{X}$, and an $n \times 1$ response vector $\mathbf{Y}$, with the properties that,

$$\begin{aligned} \mathbf{P}_{\mathbf{X}_*}\mathbf{Y} &= \widehat{\mathbf{Y}}_0, \\ (\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{Y} &= \mathbf{R}_0, \end{aligned} \tag{1}$$

where $\mathbf{P}_{\mathbf{X}_*} = \mathbf{X}_*(\mathbf{X}_*^T\mathbf{X}_*)^{-1}\mathbf{X}_*^T$, $\mathbf{X}_* = [\mathbf{1}_n : \mathbf{X}]$, $\mathbf{1}_n$ is an $n \times 1$ vector of ones, and $\mathbf{I}_n$ is an $n \times n$ identity matrix. Note that $\mathbf{X}$ does not contain a column of ones.

Consider that if a solution $(\mathbf{X}, \mathbf{Y})$ to (1) exists, then $\mathbf{R}_0^T\widehat{\mathbf{Y}}_0 = \mathbf{Y}^T(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{P}_{\mathbf{X}_*}\mathbf{Y} = 0$, that is, residual and fitted values are orthogonal. Thus, a solution exists only if

$$\mathbf{R}_0^T\widehat{\mathbf{Y}}_0 = 0. \tag{2}$$

The point is that *real* regression residuals and predicted values are orthogonal. But, because the vectors $(\widehat{\mathbf{Y}}_0, \mathbf{R}_0)$ that we are interested in *pretending are residuals and predictions* are determined by the image we want to display, it will seldom be the case that $\mathbf{R}_0^T\widehat{\mathbf{Y}}_0 = 0$. However, in Section 2.2 we describe a method to orthogonalize any image, that is, to ensure $\mathbf{R}_0^T\widehat{\mathbf{Y}}_0 = 0$, that has minor impact on the visual quality of the image. Thus, we proceed assuming that (2) holds.

2.1 Solution for $\mathbf{R}_0$ and $\widehat{\mathbf{Y}}_0$ Orthogonal

Note that there are $n(p+1)$ free "variables" in $(\mathbf{X}, \mathbf{Y})$ and only $2n$ equations in (1), thus multiple solutions exist. We provide one method of solution that allows for approximate control of the regression coefficients in the fitted model. To this end suppose that β_0 and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ are given with $|\beta_j| > 0$ for $j = 1, \ldots, p$.

Write $\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$. This looks like a statistical model, but it is not. For our purposes $(\mathbf{X}, \mathbf{Y})$ are mathematical variables whose values are to be determined so that (1) holds. The reexpression $\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ is used to transform variables from $(\mathbf{X}, \mathbf{Y})$ to $(\mathbf{X}, \boldsymbol{\epsilon})$. In terms of $(\mathbf{X}, \boldsymbol{\epsilon})$, Equations (1) are transformed to

$$\begin{aligned} \mathbf{1}_n\beta_0 + \mathbf{X}\boldsymbol{\beta} + \mathbf{P}_{\mathbf{X}_*}\boldsymbol{\epsilon} &= \widehat{\mathbf{Y}}_0, \\ (\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\boldsymbol{\epsilon} &= \mathbf{R}_0, \end{aligned} \tag{3}$$

The second equation in (3) implies that $(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{R}_0 = \mathbf{R}_0$ which in turn implies that $\mathbf{P}_{\mathbf{X}_*}\mathbf{R}_0 = \mathbf{0}_n$, or alternatively that: 1) $\mathbf{1}_n^T\mathbf{R}_0 = 0$; and 2) $\mathbf{X}^T\mathbf{R}_0 = \mathbf{0}_p$. The first equality imposes another condition, necessary when an intercept is included in the model to be fit, on the target residual vector $\mathbf{R}_0$ that is easily satisfied by centering. The second condition is satisfied provided $\mathbf{X} = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}$ for any $n \times p$ matrix $\mathbf{M}$, where $\mathbf{P}_{\mathbf{R}_0} = \mathbf{R}_0\mathbf{R}_0^T/(\mathbf{R}_0^T\mathbf{R}_0)$. Rewriting Equations (3) in terms of $(\boldsymbol{\epsilon}, \mathbf{M})$ results in

$$\begin{aligned} \mathbf{1}_n\beta_0 + (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}\boldsymbol{\beta} + \mathbf{A}_\mathbf{M}\boldsymbol{\epsilon} &= \widehat{\mathbf{Y}}_0, \\ (\mathbf{I}_n - \mathbf{A}_\mathbf{M})\boldsymbol{\epsilon} &= \mathbf{R}_0, \end{aligned} \tag{4}$$

where

$$\mathbf{A}_\mathbf{M} = \mathbf{W}(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T, \quad \text{and} \quad \mathbf{W} = [\mathbf{1}_n : (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}]. \tag{5}$$

The second equation in (4) implies that $(\mathbf{I}_n - \mathbf{A}_\mathbf{M})\mathbf{R}_0 = \mathbf{R}_0$ and $\mathbf{A}_\mathbf{M}\mathbf{R}_0 = \mathbf{0}$, which in turn implies that

$$\boldsymbol{\epsilon} = \mathbf{R}_0 + \mathbf{A}_\mathbf{M}\mathbf{Z}, \tag{6}$$

for any $n \times 1$ vector $\mathbf{Z}$. Substituting $\boldsymbol{\epsilon}$ defined in (6) into the first equation in (4) results in

$$\mathbf{1}_n\beta_0 + (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}\boldsymbol{\beta} + \mathbf{A}_\mathbf{M}\mathbf{Z} = \widehat{\mathbf{Y}}_0. \tag{7}$$

We use (7) to develop an iterative algorithm for determining $\mathbf{M}$. Let $\mathbf{M}_1, \ldots, \mathbf{M}_p$ denote the columns of $\mathbf{M}$. Manipulation of (7) shows it to be equivalent to

$$\mathbf{M}_{j_*} = \frac{1}{\boldsymbol{\beta}_{j_*}}\left\{\widehat{\mathbf{Y}}_0 - \mathbf{1}_n\beta_0 - \mathbf{A}_\mathbf{M}\mathbf{Z} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}\boldsymbol{\beta} - \left(\sum_{j=1, j\neq j_*}^{p} \boldsymbol{\beta}_j\mathbf{M}_j\right)\right\} \tag{8}$$

for any fixed j_*, $1 \le j_* \le p$ (recall that $|\beta_j| > 0$ for all j).

Equations (6) and (8) form the basis of our iterative solution. First, fix j_*, $1 \le j_* \le p$. Then compute an $n \times 1$ vector $\mathbf{Z}$, and an initial matrix $\mathbf{M}^{(0)}$ with columns $\mathbf{M}_1^{(0)}, \ldots, \mathbf{M}_p^{(0)}$. For the examples in this article the components of $\mathbf{Z}$ and $\mathbf{M}^{(0)}$ were generated as independent $\mathrm{N}(0, \tau^2)$ and $\mathrm{N}(0, \gamma^2)$ random variables respectively, with τ equal to the standard deviation of the components of $\mathbf{R}_0$, and γ equal to the standard deviation of the components of $\widehat{\mathbf{Y}}_0$. Given the current iteration $\mathbf{M}^{(k)}$, $\mathbf{M}^{(k+1)}$ is calculated column-wise as $\mathbf{M}_j^{(k+1)} = \mathbf{M}_j^{(k)}$ for $j \neq j_*$, and

$$\mathbf{M}_{j_*}^{(k+1)} = \frac{1}{\boldsymbol{\beta}_{j_*}}\left\{\widehat{\mathbf{Y}}_0 - \mathbf{1}_n\beta_0 - \mathbf{A}_{\mathbf{M}^{(k)}}\mathbf{Z} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}^{(k)}\boldsymbol{\beta} - \left(\sum_{j=1, j\neq j_*}^{p} \boldsymbol{\beta}_j\mathbf{M}_j^{(k)}\right)\right\}, \tag{9}$$

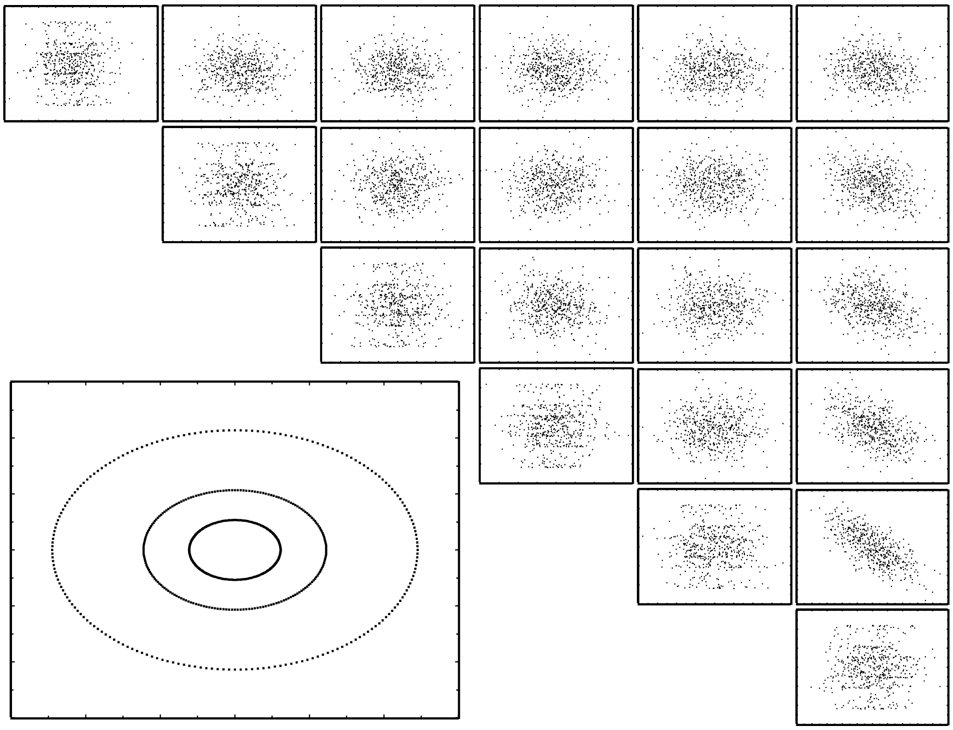


Figure 2. Bulls-eye data. Main diagonal, plots of $\mathbf{Y}$ versus $\mathbf{X}_j$, $j = 1, \ldots, 6$; upper off-diagonal; plots of $\mathbf{X}_i$ versus $\mathbf{X}_j$, $j = i + 1, \ldots, p$; lower left, plot of residuals versus predicted values.

where $\mathbf{A}_{\mathbf{M}^{(k)}}$ is calculated as in (5) with $\mathbf{M}$ replaced by $\mathbf{M}^{(k)}$. Note that only column j_* of $\mathbf{M}$ changes in the iteration. For the examples in this article convergence was declared when the maximum absolute component of $\boldsymbol{\Delta}_k = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0}) \left\{\mathbf{M}^{(k+1)} - \mathbf{M}^{(k)}\right\}$ was less than 10e − 13. The algorithm never failed to converge and usually did so in fewer than 15 iterations.

Letting $\mathbf{M}$ denote the value of $\mathbf{M}^{(k+1)}$ upon convergence, calculate $\boldsymbol{\epsilon}$ according to (6), $\mathbf{X} = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}$, and $\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$. The resulting $(\mathbf{X}, \mathbf{Y})$ are such that the least squares regression of $\mathbf{Y}$ on $[\mathbf{1}_n : \mathbf{X}]$, has predicted vector $\widehat{\mathbf{Y}}_0$, residual vector $\mathbf{R}_0$, and least squares coefficient estimates approximately equal to $(\beta_0, \boldsymbol{\beta}^T)^T$. In terms of $\mathbf{X}_* = [\mathbf{1}_n : \mathbf{X}]$, the least squares estimates $(\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}^T)^T = (\beta_0, \boldsymbol{\beta}^T)^T + (\mathbf{X}_*^T\mathbf{X}_*)^{-1}\mathbf{X}_*^T\mathbf{Z}$, where $\mathbf{Z}$ is the vector in (6). The assertion about the least squares estimates follows from (5), (6), and the fact that $\mathbf{X}_*^T\mathbf{R}_0 = \mathbf{0}_{p+1}$. Although $\mathbf{Z}$ has independent $\mathrm{N}(0, \tau^2)$ components, it is not independent of $\mathbf{X}_*$, and thus the distribution, or even the moments, of $(\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}^T)^T - (\beta_0, \boldsymbol{\beta}^T)^T$ are not simple. However, it is evident that if τ is small ($\tau \to 0$), then $(\widehat{\beta}_0, \widehat{\boldsymbol{\beta}}^T)^T - (\beta_0, \boldsymbol{\beta}^T)^T$ is small (converges to $\mathbf{0}_{p+1}$).

Finally we note that by appropriate pre-scaling of $\widehat{\mathbf{Y}}_0$, we can arrange for the model coefficient of determination to take on any desired value.

2.1.1 *Summary of Basic Algorithm*

Start with orthogonal $\widehat{\mathbf{Y}}_0$ and $\mathbf{R}_0$. In order to ensure that the correct model has coefficient of determination equal to R_0^2, redefine $\widehat{\mathbf{Y}}_0$ according to

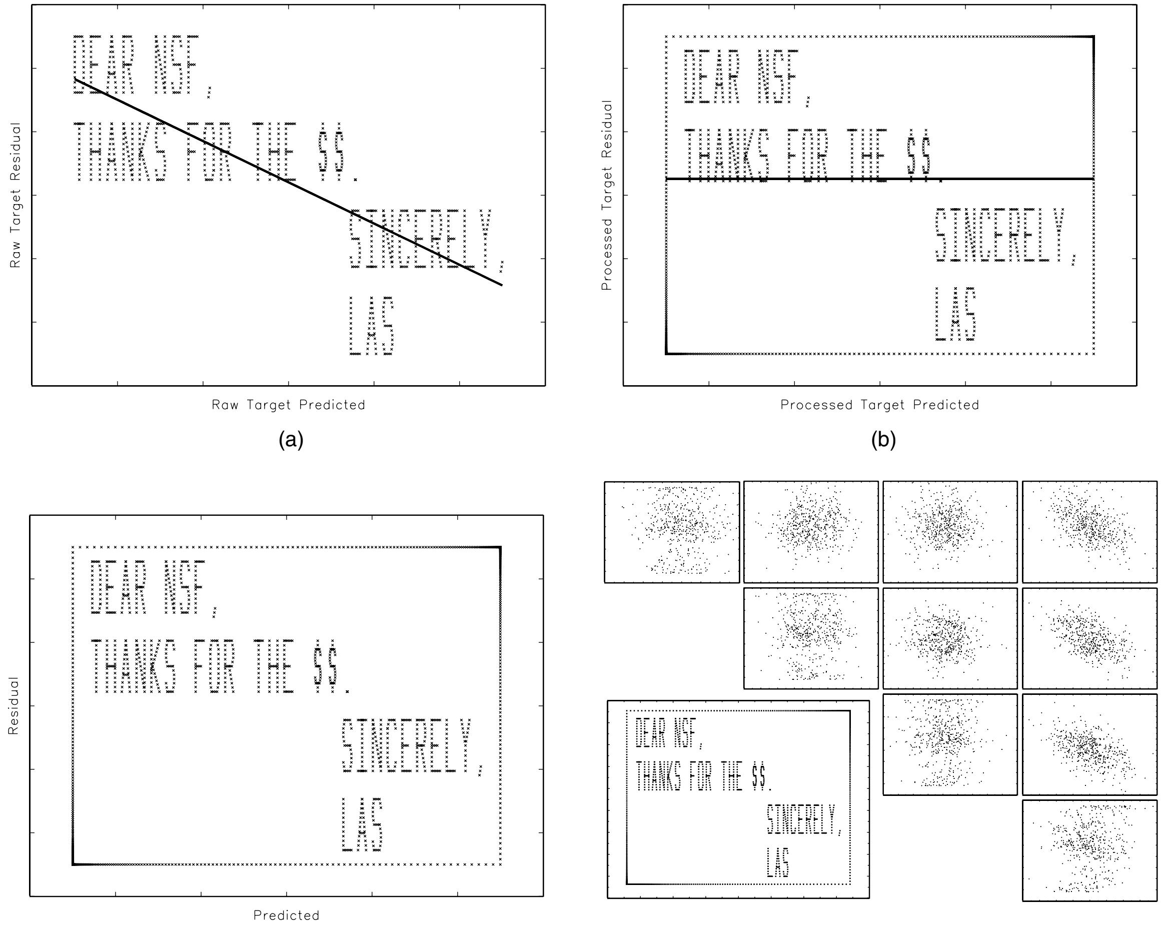
$$\widehat{\mathbf{Y}}_0 \leftarrow (s_{\mathbf{R}_0}/s_{\widehat{\mathbf{Y}}_0}) \left\{R_0^2/(1 - R_0^2)\right\}^{1/2} \widehat{\mathbf{Y}}_0,$$

where $s^2_{\mathbf{R}_0}$ and $s^2_{\widehat{\mathbf{Y}}_0}$ are the sample variances of $\mathbf{R}_0$ and the initial $\widehat{\mathbf{Y}}_0$.

1. Choose β_0 and $\boldsymbol{\beta}_{p\times 1}$ with $|\beta_j| > 0$, $j = 1, \ldots, p$;
2. Choose j_* with $1 \leq j_* \leq p$;
3. Generate $\mathbf{Z}_{n\times 1}$ with independent $\mathrm{N}(0, \tau^2)$ components;
4. Generate $\mathbf{M}^{(0)}_{n\times p}$ with independent $\mathrm{N}(0, \gamma^2)$ components;
5. Iterate $\mathbf{M}^{(k)}$ according to (9) until convergence denoting the final iterate by $\mathbf{M}$;
6. Calculate $\boldsymbol{\epsilon}$ according to (6);
7. Calculate $\mathbf{X} = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}$, and $\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$.

2.1.2 *Example: Bull's Eye Data*

For this example 600 points (t_i, w_i) lying on three concentric circles were calculated. Thus, the scatterplot of ordinates $\mathbf{w}$ versus abscissae $\mathbf{t}$ exhibits a target or bull's eye pattern. We take $\mathbf{R}_0 = \mathbf{w}$ and $\widehat{\mathbf{Y}}_0 = (s_w/s_t)\left\{R_0^2/(1 - R_0^2)\right\}^{1/2}\mathbf{t}$, where s_w^2 and s_t^2 are the sample variances of $\mathbf{w}$ and $\mathbf{t}$, and the target coefficient of determination was $R_0^2 = .75$. The orthogonality condition is satisfied via the circular symmetry, and the scaling in the calculation of $\widehat{\mathbf{Y}}_0$ ensures that the constructed data will have least squares coefficient of determination $R_0^2 = .75$. Data $(\mathbf{X}, \mathbf{Y})$ were generated using the algorithm in Section 2 with $p = 6$, $\beta_0 = 0$, $\boldsymbol{\beta} = (1, 2, 3, 4, 5, 6)^T$, and $j_* = 6$. Convergence was achieved in eight iterations. The least squares coefficient estimates, including the intercept, are (−0.03, 0.97, 2.01, 2.96, 3.91, 4.91, 5.89).



(a)

(b)

(c)

(d)

Figure 3. Figure (a), nonorthogonal target message with least squares line superimposed; (b) orthogonalization, as evidence by the least squares line, is achieved by the addition of clustered frame points; (c) orthogonalized message without least squares line; (d) plots of the data from a four-predictor dataset created from the image.

The data and the residual plot are displayed in Figure 2. Plots of **Y** versus each of the six predictors appear along the main diagonal, whereas plots of predictor pairs appear along the off diagonal. The plot of residual versus predicted values appears in the lower left. The embedded-image datasets typically are large, that is, n is large. Thus, for the data plots in Figure 2 and in all subsequent figures, only min(n, 500) randomly selected data points are plotted. However, the residual plots necessarily display all n data points. Note that there is little evidence in the data plots of the unusual residual-predictor pattern. Pairwise predictor collinearity is not visually noteworthy. Also the nonintercept-adjusted and intercepted-adjusted condition numbers of the scaled $\mathbf{X}^T\mathbf{X}$ matrix are 5.06 and 4.32, respectively, indicating little multicollinearity.

2.2 Nonorthogonal Residual and Prediction Vectors

Figure 3(a) displays the graph of a message to be embedded in a dataset. Superimposed on the message is the least squares line obtained by regressing the ordinate on the abscissa. As is, these points do not satisfy the orthogonality condition (2). (Hopefully the negative slope is not indicative of future NSF funding levels!) However, the least squares line suggests a remedy. Adding leverage points to the upper-right and lower-left corner of the plots would rotate the least squares line toward the horizontal. In fact, it is a reasonable graduate-level exercise to determine analytically two corner locations and the numbers of points needed at each location so that the least squares line has slope exactly equal to zero. This solution works, but has the undesirable and unnatural feature of creating datasets with numerous replicate observations at two points.

Figure 3(b) illustrates a more palatable solution. Note the frame around the message, and especially notice that the points framing the message have higher concentrations in the upper-right and lower-left corners. The nonuniform concentration of frame points is determined precisely so that the least squares line fit to the new set of points (the original message points and the frame points) has zero slope, thus ensuring the orthogonality required by (2). This is accomplished by using a *parametric frame* of points where the parameter controls the amount of clustering to either set of opposing corners. Figure 3(c) shows the message with the frame but without the least squares line overlaid. Figure 3(d) displays plots of the data from a four-variable dataset constructed using the algorithm in Section 2 having the framed message as its embedded residual plot.

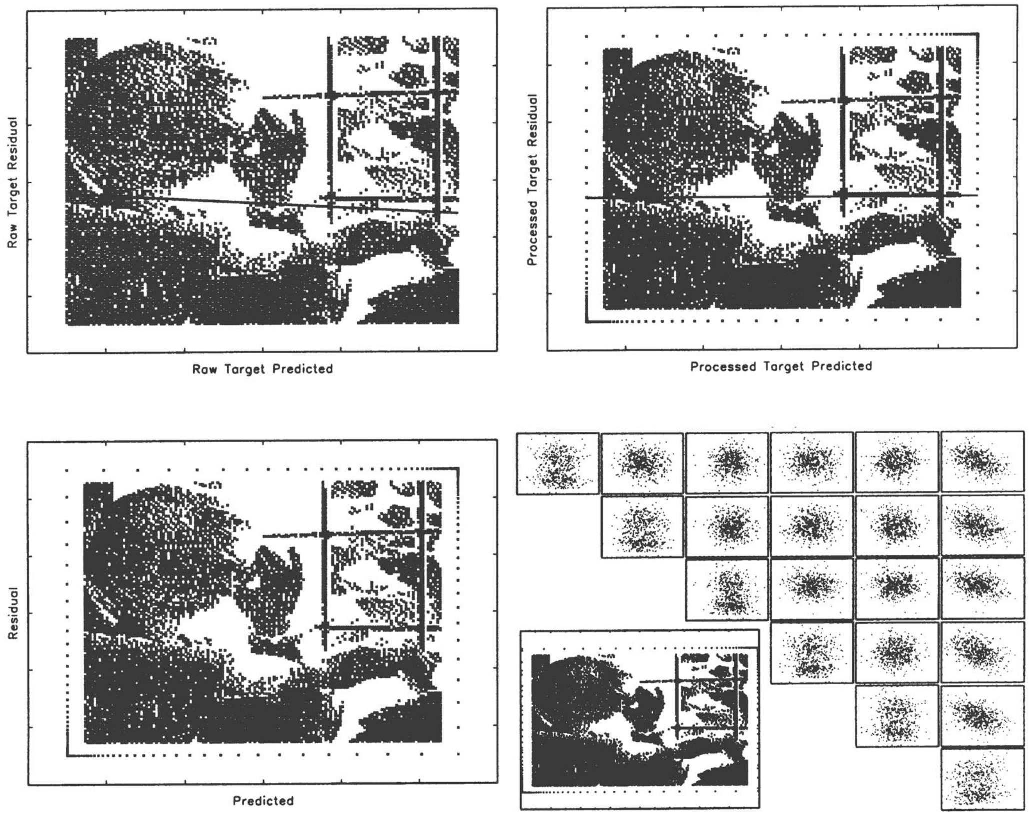


Figure 4. Figure (a), nonorthogonal target image with least squares line superimposed; (b) orthogonalization, as evidence by the least squares line, is achieved by the addition of clustered frame points; (c) orthogonalized image without least squares line; (d) plots of the data from a six-predictor dataset created from the image.

The parametric frame is constructed as follows. Suppose that the four points $(x_{\min}, y_{\min})$, $(x_{\min}, y_{\max})$, $(x_{\max}, y_{\max})$, $(x_{\max}, y_{\min})$ with $x_{\min} < x_{\max}$ and $y_{\min} < y_{\max}$ define the corners of a box that frames the message/image of interest. A parametric frame is created by taking a set of equally spaced points $0 = u_1 < u_2 < \cdots < u_{m-1} < u_m = 1$ and raising them to a power $\alpha > 0$. The lower edge of the frame has abscissae given by $x_{\min} + (x_{\max} - x_{\min})u_j^{\alpha}$ and common ordinates $y_{\min}$. The upper edge of the frame has abscissae given by $x_{\min} + (x_{\max} - x_{\min})(1 - u_j^{\alpha})$ and common ordinates $y_{\max}$. The left edge has common abscissae $x_{\min}$ and ordinates $y_{\min} + (y_{\max} - y_{\min})u_j^{\alpha}$; and the right edge has common abscissae $x_{\max}$ and ordinates $y_{\min} + (y_{\max} - y_{\min})(1 - u_j^{\alpha})$. These frame points are appended to the message/image data of interest. When $\alpha = 1$ the points on the frame are equally spaced along the segments on which they lie. As $\alpha \to 0$ equal numbers of points converge to the upper-left and lower-right corners; whereas as $\alpha \to \infty$ equal numbers of points converge toward the lower-left and upper-right corners. Thus, by taking m large enough to ensure sufficient leverage, varying α over $(0, \infty)$ can change the slope of a regression line fit to the totality of the points continuously from negative to positive. A computer program can be written to solve for the particular value of α yielding zero slope. This is how the frame points were determined for Figure 3(b), also displayed in Figure 3(c) without the least squares line superimposed.

Some images look better with a fuzzy frame, see for example Figure 1(b)–(f). Fuzzy frames are created by replacing the constant abscissae and ordinates in the rigid-frame construction with random vectors. For example, the lower edge of a fuzzy frame still has abscissae given by $x_{\min} + (x_{\max} - x_{\min})u_j^{\alpha}$, $j = 1, \ldots, m$, but the common ordinates $y_{\min}$ are replaced by an $m \times 1$ vector of independent $\mathrm{N}(y_{\min}, \eta^2)$ random variables where η controls the fuzziness.

The parametric framing strategy works with images as well. Figure 4(a) displays the author's black-and-white rendition of a famous image of R. A. Fisher working at a hand calculator. The overlaid least squares line again reveals the lack of orthogonality between the raw image's ordinate and abscissa vectors. The addition of parametric frame points calculated as described above induces orthogonality as is evident in Figure 4(b) and (c). Finally, Figure 4(d) displays plots of the data from a six-variable dataset constructed using the algorithm in Section 2 having the framed image as its embedded residual plot.

3. REMARKABLE RESIDUALS REDUX

We now show a few additional residual plots of general interest. Our intent is to illustrate a range of possibilities to help stimulate readers making their own plots.

Although often too much emphasis is placed on normality of the equation errors in multiple linear regression, students are nevertheless trained to check for departures from normality by examining residuals. Imagine the reaction of a student, having been so instructed to check for normality of the errors, to the residual

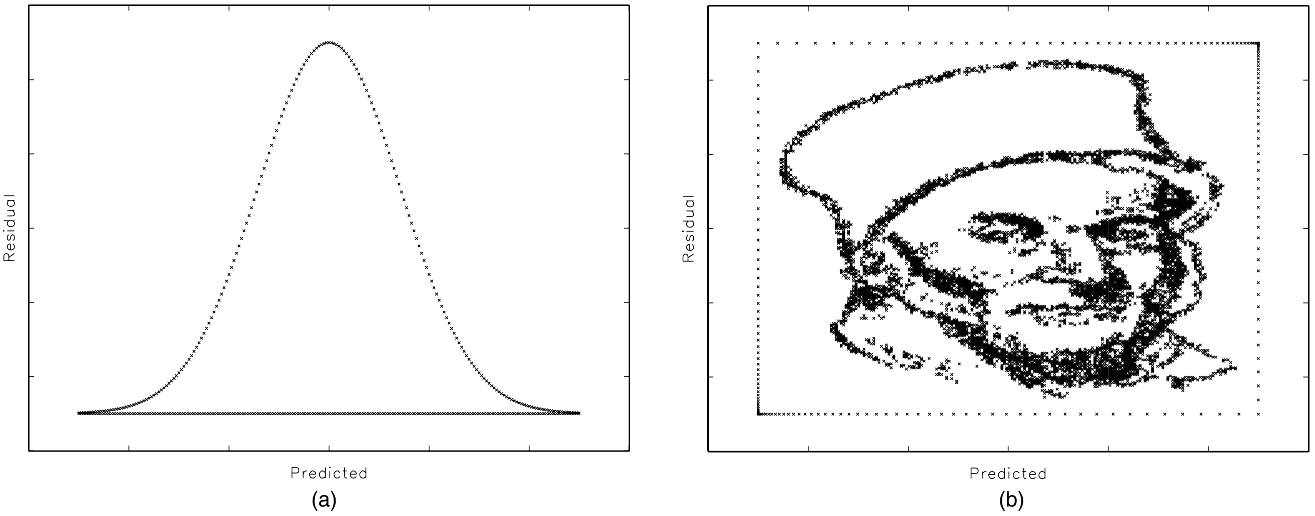


Figure 5. Two Gaussian residual plots.

plot in Figure 5(a)—looks pretty normal to me! Of course, it is difficult to imagine any residual plot being more Gaussian (Carl Friedrich Gauss(ian), that is) than the one in Figure 5(b).

Note the lack of a frame in Figure 5(a). A frame is not necessary due to the symmetry of the normal density. The image in Figure 5(b) is the author's rendition of a well-publicized drawing of Carl Friedrich Gauss. Comparison of Figure 5(b) to the popular drawing of Gauss shows that the former has much less black area translating to far fewer observations in the constructed dataset (smaller *n*). Image datasets can be quite large and often some preprocessing with an image-editing software program is required to reduce the datasets to a manageable size; see the Appendix (p. 175).

The Department of Statistics at NCSU has a long (65 years and counting) and storied past. Although the reports that Patterson Hall is haunted have never been verified, it is likely that statistics students at North Carolina State University would conclude that the ghosts of statistics present and past were astir if either of the residual plots in Figure 6 appeared on their computer monitor.

Some statisticians have strong prior opinions about what to expect from data. Perhaps such statisticians would not be at all surprised to see residual plots like those in Figure 7 appear on their computer screens.

The residual plots in Figure 8 will catch students off guard initially, but closer inspection will reveal something fishy about them. However, contrary to first impressions, these are not residual plots from a Poisson regression model.

Famous quotations about statistics make good fodder for hidden messages. In addition to their entertainment value, they complement the professional statistician's academic training. Two such quotes are shown in the residual plots in Figure 9. The origins of the "Damn Lies" quotation has never been firmly established, but the author has recently obtained new evidence pointing toward the former republican president. (Although many would dispute attribution to the former president, there is widespread sentiment that the essence of the quotation was a central tenet of *Reaganomics*.) In addition to offering sound advice to anyone interpreting statistics, the second quotation also

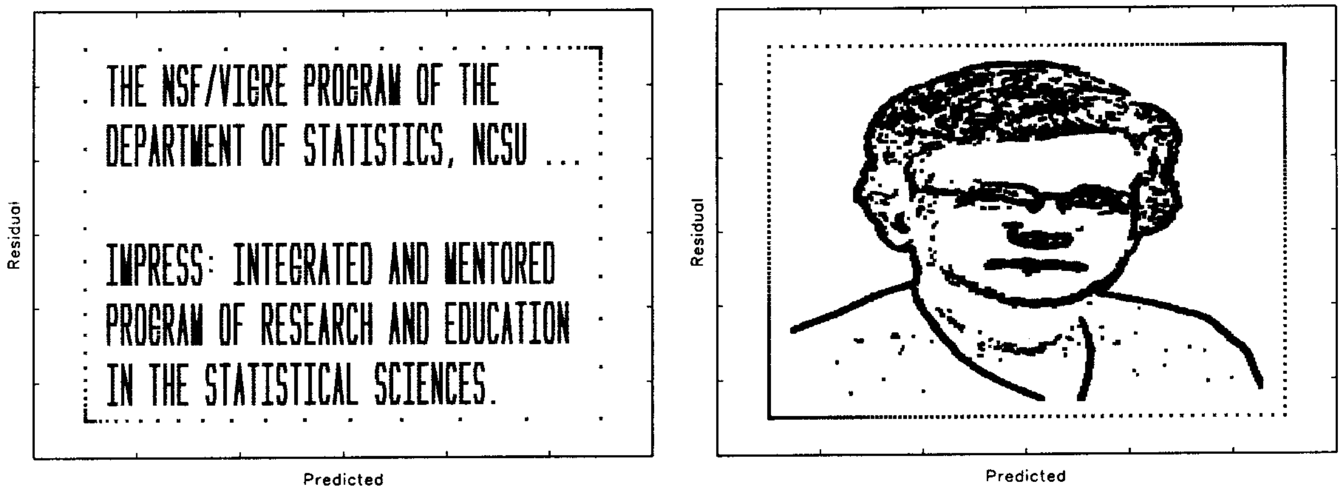


Figure 6. The National Science Foundation's VIGRE program has done much to shape departmental life at NCSU in recent years. However, the core philosophy of the department dates back to its origins, and to the leadership provided by its first head, Gertrude Cox.

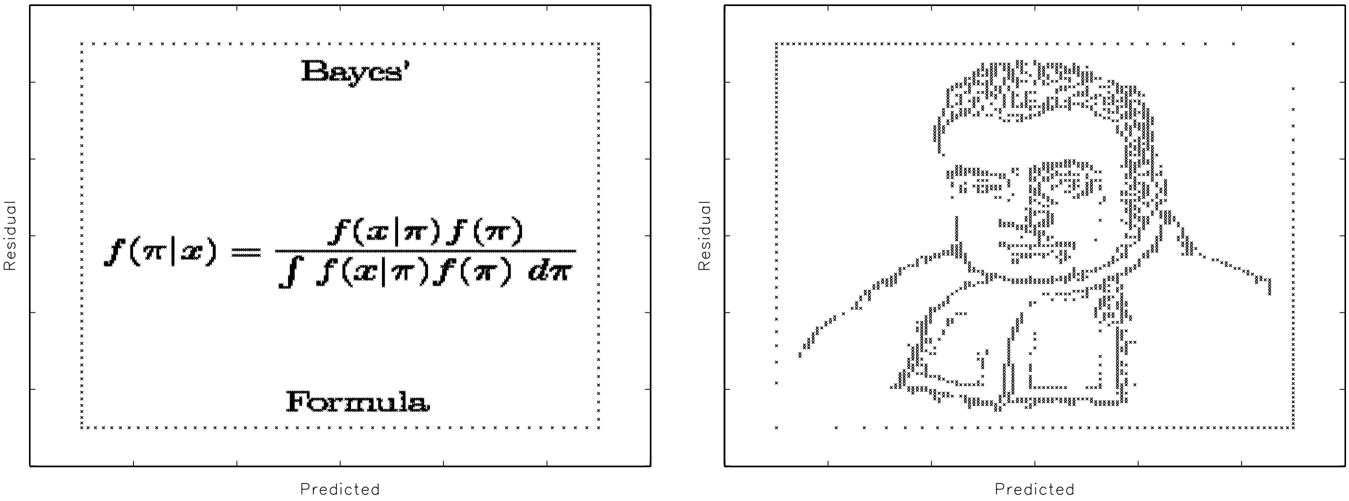


Figure 7. Two preposterous posterior residual plots.

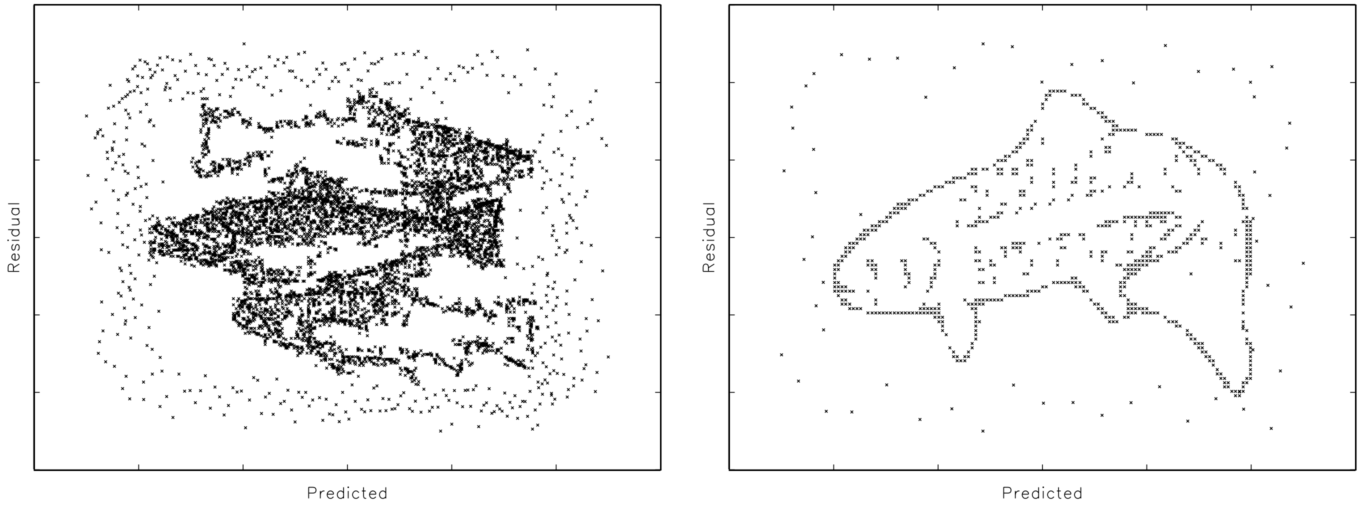


Figure 8. Two fishy residual plots.

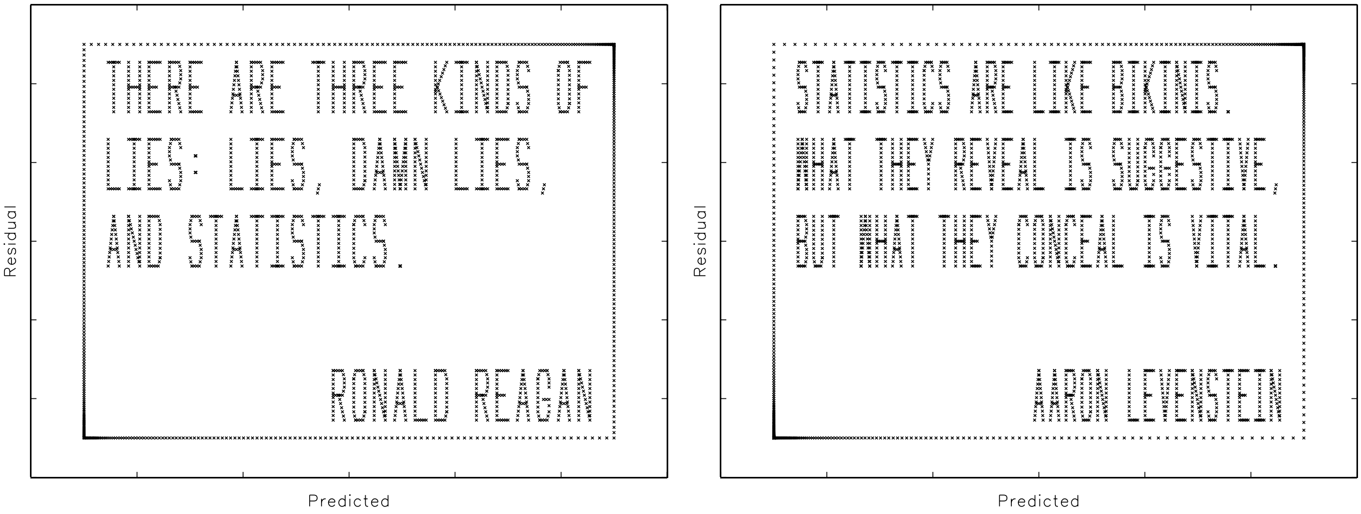


Figure 9. Two famous quotations about statistics.

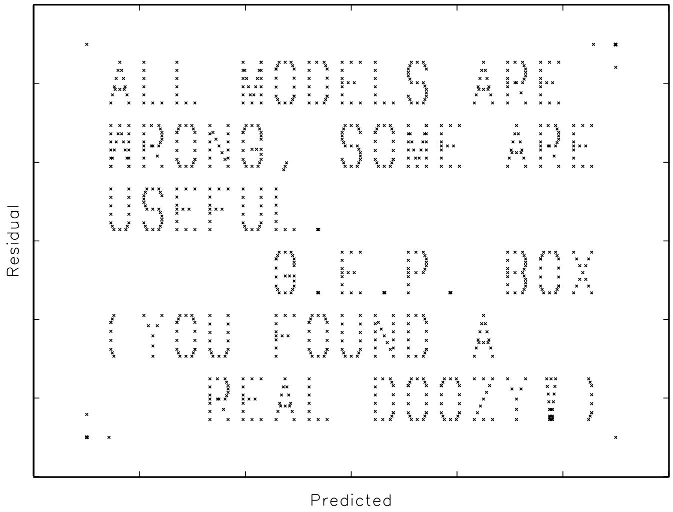


Figure 10. Model building wisdom from G.E.P. Box.

suggests that the study of "vital statistics" is one of the more interesting subfields of our discipline. (The sensitive reader should bear (not bare!) in mind that bikinis are worn by both women and men.)

4. VARIABLE SELECTION PROPERTIES

The embedded-image datasets can be used for entertaining examples as is, or additional uninformative variables can be appended to the set of explanatory variables and then used for variable-selection exercises. The difficulty of identifying the correct set of predictors from among a large set of candidate predictors is largely dependent on the number of uninformative variables and the strength of the relationship between the response and the informative variables. Both of these factors are easily controlled. The data-construction algorithm in Section 2 allows the correct-model coefficient of determination, R_0^2, to be specified, and we can generate as many uninformative predictor variables as we want. We now illustrate the construction of suitable datasets with an example. Section 5.1 (p. 173) contains a related example.

4.1 Box's Model Quote

Two six-variable datasets were constructed using the algorithm in Section 2 having the residual plot in Figure 10. One with $R_0^2 = 0.15$, the other with $R_0^2 = 0.90$. Next 100 uninformative predictors were generated and appended to each set of six informative predictors. Then the columns of the predictor matrices were randomly permuted to obscure the positions of the informative variables.

We ran forward and backward selection with slentry/slstay= .05, .01, and .001 on both datasets. A summary of the results is apparent in the residual plots of the selected models in Figure 11. For the dataset with the weaker signal ($R_0^2 = 0.15$), forward selection with slentry = .01, FS(.01), identifies the correct six-variable model, whereas the FS(.05) model is too large (nine variables) and the FS(.001) is too small (two variables). The backward selection model with slstay = .05, BS(.05), is too large (nine variables), but both the BS(.01) and BS(.001) identify the correct model. Model identification for the dataset with the stronger signal ($R_0^2 = .90$) should be easier and this is the case. With 100 uninformative variables it's not surprising that slentry/slstay = .05 allows some uninformative variables to be selected. But with $R_0^2 = .90$ all of the informative variables are highly significant and thus even FS(.001) does not exclude any of them.

With a little trial and error it is often possible to find a value of R_0^2 such that BS(.05) is too large, BS(.001) is too small, and BS(.01) is just right (note that Figure 11 already establishes that this is possible for FS($\cdot$)). In fact, for the Box quote residual plot, setting $R_0^2 = .033$ in the data construction algorithm produces a dataset with these properties. The relevant point is, that by varying the number of uninformative variables appended to the dataset and R_0^2, it is possible to construct datasets such that a cursory application of forward or backward selection will not reveal the embedded image, but a more creative use of these tools will be successful. Of course other variable selection methods can be used, we consider only forward and backward selection here for simplicity and their accessibility to undergraduates.

5. DATASETS FOR SECOND-ORDER MODELS

The dataset construction algorithm in Section 2 does not allow for models with powers and cross-product terms. We now show how to adapt the basic algorithm to include such models. We consider a specific quadratic model, but the strategy applies more generally. Suppose that we want to embed an image in the residual plot of a "correct" model of the form

$$\begin{aligned} &E(Y \mid X_1^*, X_2^*, X_3^*, \ldots, X_{p^*}^*) \\ &\quad = \beta_0^* + \beta_1^* X_1^* + \beta_2^*(X_1^*)^2 + \beta_3^* X_2^* + \beta_4^*(X_2^*)^2 \\ &\qquad + \beta_5^*(X_1^* X_2^*) + \beta_6^* X_3^* + \cdots + \beta_{p^*}^* X_{p^*}, \end{aligned} \tag{10}$$

where $p^* \geq 3$ and all of the variables in (10) are scalars. The superscript "$*$" in (10) is used to distinguish between the variables in (10) and those in the matrix version of the linear model formulation of this model. We write the latter as

$$E(\mathbf{Y} \mid \mathbf{X}) = \mathbf{1}_n \beta_0 + \mathbf{X}\boldsymbol{\beta}, \tag{11}$$

where $\mathbf{X}$ is $n \times p$ with $p = p^* + 3$ and has columns $\mathbf{X}_1, \ldots, \mathbf{X}_p$ such that

$$\mathbf{X}_2 = (\mathbf{X}_1 \bullet \mathbf{X}_1), \quad \mathbf{X}_4 = (\mathbf{X}_3 \bullet \mathbf{X}_3), \quad \mathbf{X}_5 = (\mathbf{X}_1 \bullet \mathbf{X}_3), \tag{12}$$

where "$\mathbf{A} \bullet \mathbf{B}$" denotes the Hadamard (element-wise) product of the vectors $\mathbf{A}$ and $\mathbf{B}$.

As in Section 2 our goal is to determine $(\mathbf{X}, \mathbf{Y})$ satisfying the conditions in (1). The difference is that we now have additional constraints on the columns of $\mathbf{X}$. It is still the case that $\mathbf{X} = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}$ for some matrix $\mathbf{M}$, that is, $\mathbf{X}_j = (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_j$, $j = 1, \ldots, p$. In terms of the columns of $\mathbf{M}$ the constraints (12) are:

$$\begin{aligned} (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_2 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\Big\}; \\ (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_4 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\}; \\ (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_5 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\}. \end{aligned} \tag{13}$$

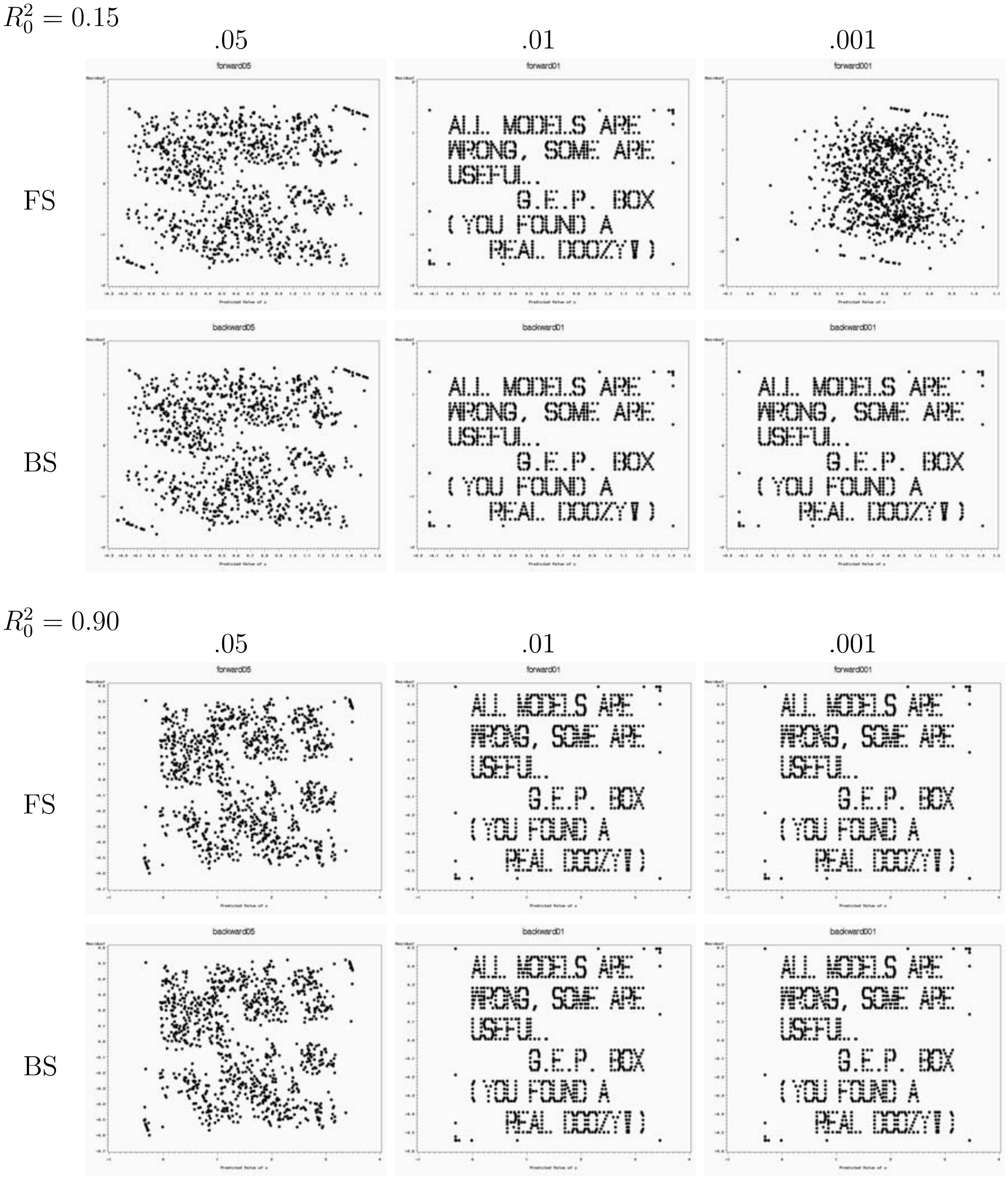


Figure 11. Box quote datasets selected-models residual plots. FS, forward selection; BS, backward selection; .05, .01,. 001 slentry/slstay for forward/backward selection; R_0^2, correct model coefficient of determination.

The equations in (13) are equivalent to:

$$\begin{aligned}\mathbf{M}_2 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_2;\\ \mathbf{M}_4 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_4;\\ \mathbf{M}_5 &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_5.\end{aligned} \tag{14}$$

The latter equations suggest some simple modifications of the algorithm in Section 2.1.1. In Step 2, take $j_* \geq 6$, and in Step 5, in addition to the updating formula in (9), add the following updating formulas:

$$\begin{aligned}\mathbf{M}_2^{(k+1)} &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_2^{(k)};\\ \mathbf{M}_4^{(k+1)} &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_4^{(k)};\\ \mathbf{M}_5^{(k+1)} &= (\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\Big\{\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_1\big]\bullet\big[(\mathbf{I}_n - \mathbf{P}_{\mathbf{R}_0})\mathbf{M}_3\big]\Big\} + \mathbf{P}_{\mathbf{R}_0}\mathbf{M}_5^{(k)}.\end{aligned} \tag{15}$$

The other steps of the algorithm remain unchanged.

This modified algorithm is fast and reliable but has one drawback. The numerical error in the computation of the derived-variable columns, $\mathbf{X}_2, \mathbf{X}_4$, and $\mathbf{X}_5$, is great enough to distort the intended residual image plot when these columns are replaced by columns recalculated from the base variables $\mathbf{X}_1$ and $\mathbf{X}_3$. In other words, the algorithm produces a response vector $\mathbf{Y}$ and a design matrix $\mathbf{X} = [\mathbf{X}_1 : \mathbf{X}_2 : \cdots : \mathbf{X}_p]$ with the property that the residual plot from the regression of $\mathbf{Y}$ on $[\mathbf{1}_n : \mathbf{X}]$ exhibits the intended image. However, the discrepancies in

$$\mathbf{X}_2 - (\mathbf{X}_1 \bullet \mathbf{X}_1), \quad \mathbf{X}_4 - (\mathbf{X}_3 \bullet \mathbf{X}_3), \quad \mathbf{X}_5 - (\mathbf{X}_1 \bullet \mathbf{X}_3), \tag{16}$$

although small, are nevertheless large enough that the residual plot from the regression of $\mathbf{Y}$ on $[\mathbf{1}_n, \mathbf{X}^\dagger]$ where

$$\begin{aligned}\mathbf{X}^\dagger = [\mathbf{1}_n : \mathbf{X}_1 : (\mathbf{X}_1 \bullet \mathbf{X}_1) :&\\ \mathbf{X}_3 : (\mathbf{X}_3 \bullet \mathbf{X}_3) : (\mathbf{X}_1 \bullet \mathbf{X}_3) : \mathbf{X}_6 : \cdots : \mathbf{X}_p]&\end{aligned} \tag{17}$$

is distorted unacceptably. This is relevant, because in a model-finding exercise, one would typically provide to students only the base variables $\mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p$ and expect them to construct the derived variables (squares and cross products) as part of the model-building exercise. Thus, we want to ensure the residual plot $\mathbf{Y}$ on $[\mathbf{1}_n, \mathbf{X}^\dagger]$ is distortion free. We accomplish this with a second level of numerical fine tuning.

Recall that our objective, in terms of $\mathbf{X}^\dagger$, is to have

$$\mathbf{P}_{\mathbf{X}_*}\mathbf{Y} = \widehat{\mathbf{Y}}_0, \quad \text{and} \quad (\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{Y} = \mathbf{R}_0, \tag{18}$$

where now $\mathbf{P}_{\mathbf{X}_*} = \mathbf{X}_*(\mathbf{X}_*^T\mathbf{X}_*)^{-1}\mathbf{X}_*^T$ and $\mathbf{X}_* = [\mathbf{1}_n : \mathbf{X}^\dagger]$. Alternatively we seek $\mathbf{Y}$ and $(\mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p)$ to minimize

$$\begin{aligned}&Q^*(\mathbf{Y}, \mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p)\\ &\quad = \|\mathbf{P}_{\mathbf{X}_*}\mathbf{Y} - \widehat{\mathbf{Y}}_0\|^2 + \|(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{Y} - \mathbf{R}_0\|^2.\end{aligned} \tag{19}$$

In fact, at its absolute minimum $Q^* = 0$; however, we need only ensure that it is acceptably small. For fixed $(\mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p)$, Q^* is a quadratic function of $\mathbf{Y}$ and is minimized at $\mathbf{Y} = (\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{R}_0 + \mathbf{P}_{\mathbf{X}_*}\widehat{\mathbf{Y}}_0$. Thus, we seek to minimize

$$\begin{aligned}&Q(\mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p)\\ &\quad = Q^*\Big((\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{R}_0 + \mathbf{P}_{\mathbf{X}_*}\widehat{\mathbf{Y}}_0, \mathbf{X}_1, \mathbf{X}_3, \mathbf{X}_6, \ldots, \mathbf{X}_p\Big)\\ &\quad = \mathbf{R}_0^T\mathbf{P}_{\mathbf{X}_*}\mathbf{R}_0 + \widehat{\mathbf{Y}}_0^T(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\widehat{\mathbf{Y}}_0.\end{aligned} \tag{20}$$

So the two-stage algorithm we use to construct quadratic models with embedded residual plots starts with the calculation of initial values $\widetilde{\mathbf{Y}}$ and $\widetilde{\mathbf{X}}$ using the modifications of Steps 2 and 5 of the basic algorithm (in Section 2.1.1) described above. Then extract the base columns $\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_3, \widetilde{\mathbf{X}}_6, \ldots, \widetilde{\mathbf{X}}_p$ from $\widetilde{\mathbf{X}}$ and use these as starting values in the minimization of Q. For the examples in this article Q was minimized by successively minimizing it with respect to the ith row of the matrix $[\mathbf{X}_1 : \mathbf{X}_3 : \mathbf{X}_6 : \ldots : \mathbf{X}_p]$, $i = 1, \ldots, n$, and recycling through rows as needed until $Q < 10^{-8}$.

For large n the latter sequential minimization can be time consuming in the absence of good starting values. However, good starting values can been obtained by running the modified basic algorithm several times using different random $\mathbf{Z}$ and $\mathbf{M}^{(0)}$ matrices and calculating the overall discrepancy measure

$$\begin{aligned}D = \|\widetilde{\mathbf{X}}_2 - (\widetilde{\mathbf{X}}_1 \bullet \widetilde{\mathbf{X}}_1)\|^2&\\ + \|\widetilde{\mathbf{X}}_4 - (\widetilde{\mathbf{X}}_3 \bullet \widetilde{\mathbf{X}}_3)\|^2 + \|\widetilde{\mathbf{X}}_5 - (\widetilde{\mathbf{X}}_1 \bullet \widetilde{\mathbf{X}}_3)\|^2&\end{aligned} \tag{21}$$

for each random start, see Equation (16). Then choose as the initial starting values those $\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_3, \widetilde{\mathbf{X}}_6, \ldots, \widetilde{\mathbf{X}}_p$ associated with the minimum value of D.

5.1 Example: Correct Model Message

Figure 12 displays residual plots embedded in a model fully quadratic in two variables (five terms) and linear in four other variables. This is the case described above with $p = 9$. The upper-left and -right panels show the residual plot before and after the fine tuning algorithm described above. The fine-tuning algorithm is not perfect as is evident by the small perturbations remaining in the rectangular parametric frame. The lower panel displays plots of the dataset base variables.

The algorithm produces a response vector $\mathbf{Y}$ and six explanatory-variables columns $\mathbf{X}_1, \ldots, \mathbf{X}_6$, and the "correct model" for this dataset is fully quadratic in the first two variables and linear in the latter four. The mismatch between the actual model and the correct model identified by the residual plot is resolved once these generated informative predictor variables are mixed among other uninformative variables creating a greater challenge for model finding methods. For example, suppose that five additional uninformative predictor variables, $\mathbf{T}_1, \ldots, \mathbf{T}_5$ are generated independently of $\mathbf{Y}$. Now put all of the variables in a new predictor matrix

$$\mathbf{X}_{\text{New}} = [\mathbf{X}_3 : \mathbf{X}_4 : \mathbf{X}_1 : \mathbf{X}_5 : \mathbf{T}_1 : \mathbf{T}_2 : \mathbf{X}_6 : \mathbf{T}_3 : \mathbf{X}_2 : \mathbf{T}_4 : \mathbf{T}_5],$$

in the order indicated. Then the correct model for the data $(\mathbf{X}_{\text{New}}, \mathbf{Y})$ is the model in the residual plot. The challenge, of course, is identifying the correct model terms.

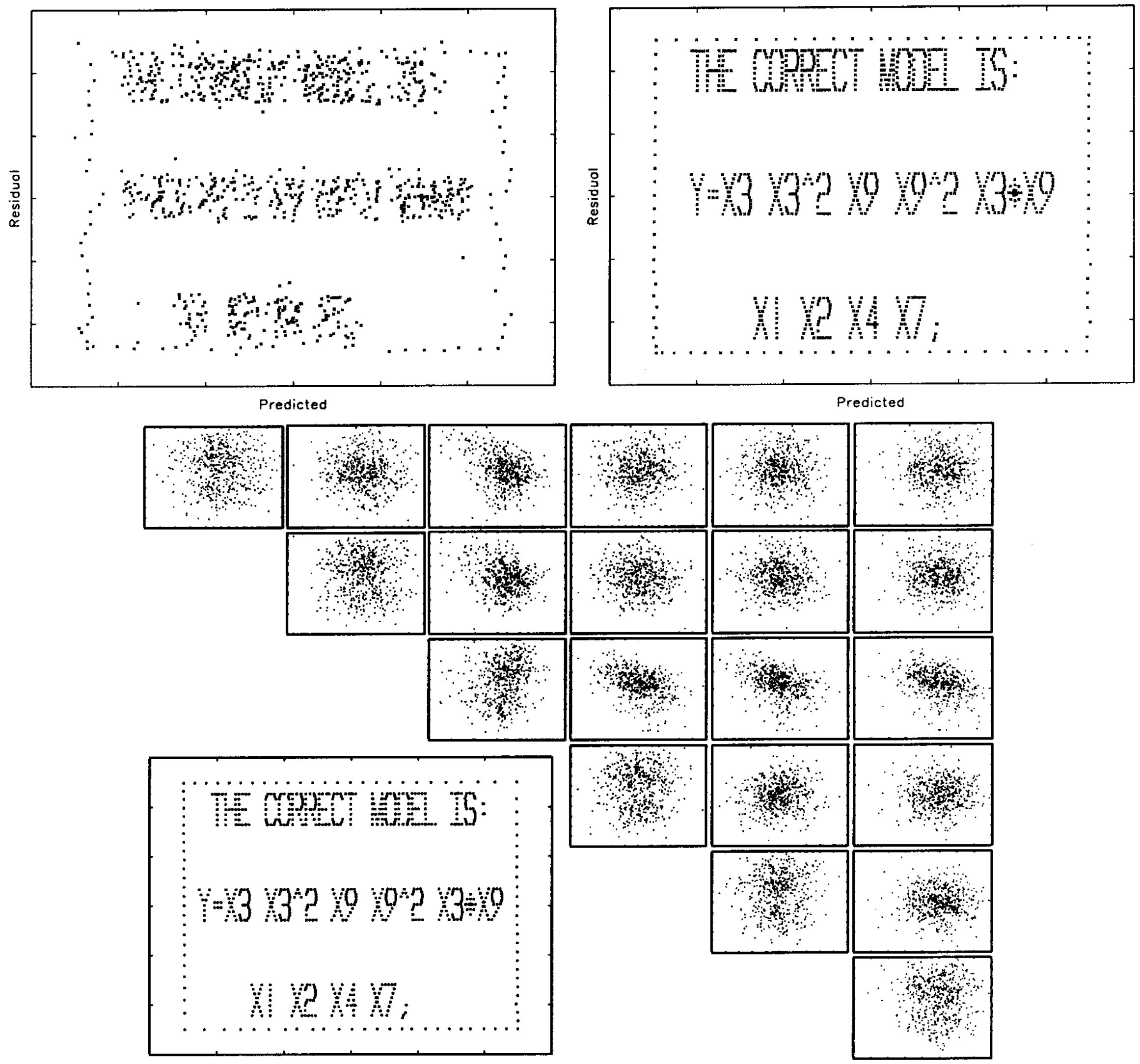


Figure 12. Residual plots from quadratic models of the form $y = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_2 + \beta_4 x_2^2 + \beta_5 x_1 x_2 + \beta_6 x_3 + \beta_7 x_4 + \beta_8 x_5 + \beta_9 x_6$. The upper left and right panels are residual plots before (left) and after (right) the fine tuning algorithm described in Section 5. The lower panel displays plots of the data: response versus predictor on the main diagonal; predictor versus predictor on the off diagonals. Only base variables are included.

Table 1 compares the results of forward and backward selection applied to the 11 base variables in $\mathbf{X}_{\text{New}}$, their squares, and the 55 possible cross-product terms (77 total predictors). Forward selection (slentry $=$ .01) identified an inclusive 14-variable model. The backward-selection (slstay $=$ .01) model differed from the forward-selection model in that it contained one fewer uninformative variable.

With slentry and slstay reduced to .001, backward selection nails the correct model whereas forward selection finds an inclusive 11-variable model containing two uninformative variables. However, the two superfluous variables have p values $>$ .3 and thus are apparent candidates for elimination resulting in the "correct" model.

Just as with the Box quotation example in Section 4.1, this example dataset is such that whereas a cursory application of forward or backward selection will not reveal the hidden message, a deeper probing of the data using these basic variable selection tools will reveal the "correct" model, thus rewarding the conscientious student.

6. IS THERE A CORRECT MODEL?

Because the datasets constructed by the algorithms in Sections 2 and 5 are not generated from a usual linear statistical model, it begs the question of what is meant by a "correct" model in the discussions of variable selection in Sections 4 and 5. Suppose that data $(\mathbf{X}, \mathbf{Y})$ are generated via one of the algorithms in Section 2 and 5. Now suppose that additional variables $\mathbf{T}$ are appended to the predictor matrix. The least squares estimate of the coefficient vector of $\mathbf{T}$ in the linear model containing both $\mathbf{X}$ and $\mathbf{T}$ is

$$\begin{aligned}\widehat{\boldsymbol{\beta}}_T &= \left\{\mathbf{T}^T(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{T}\right\}^{-1}\mathbf{T}^T(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{Y} \\ &= \left\{\mathbf{T}^T(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{T}\right\}^{-1}\mathbf{T}^T\mathbf{R}_0, \qquad (22)\end{aligned}$$

where $\mathbf{P}_{\mathbf{X}_*}$ is the projection matrix of $\mathbf{X}_* = [\mathbf{1}_n : \mathbf{X}]$. The second equality, which follows from (1), makes clear that $\widehat{\boldsymbol{\beta}}_T$ is not a linear function of uncorrelated, homoscedastic errors, as it would be if the data were generated according to usual linear model assumptions. However, if the randomly generated uninformative

Table 1. Variable selection method comparison for the "Correct Model" dataset. Inf., informative; Un., uninformative; FS(α), forward selection with slentry $= \alpha$; BS(α), forward selection with slstay $= \alpha$. Table entries are p values of the final models identified by forward and backward selection; "—", variable not selected.

Variable		*Selection Method*			
Inf.	Un.	FS(.01)	BS(.01)	FS(.001)	BS(.001)
x1		< .0001	< .0001	< .0001	< .0001
x2		< .0001	< .0001	< .0001	< .0001
x3		< .0001	< .0001	< .0001	< .0001
x4		< .0001	< .0001	< .0001	< .0001
x7		< .0001	< .0001	< .0001	< .0001
x9		< .0001	< .0001	< .0001	< .0001
	x1*x1	0.1253	0.0042	0.3266	—
x3*x3		< .0001	< .0001	< .0001	< .0001
x9*x9		< .0001	< .0001	< .0001	< .0001
	x1*x3	0.1694	—	0.4740	—
	x2*x9	0.0054	0.0030	—	—
	x3*x7	0.0020	0.0051	—	—
x3*x9		< .0001	< .0001	< .0001	< .0001
	x8*x9	0.0089	0.0090	—	—

predictors $\mathbf{T}$ are such that $-\mathbf{T}$ and $\mathbf{T}$ are equal in distribution given $\mathbf{X}$, then $\widehat{\boldsymbol{\beta}}_T$ has mean zero under repeated sampling of $\mathbf{T}$ by virtue of the fact that $\widehat{\boldsymbol{\beta}}_T$ is an odd function of $\mathbf{T}$, that is, $\widehat{\boldsymbol{\beta}}_T(-\mathbf{T}) = -\widehat{\boldsymbol{\beta}}_T(\mathbf{T})$. In this sense, $\widehat{\boldsymbol{\beta}}_T$ does have mean $\mathbf{0}$. Of course, the sampling theory assumed by variable-selection methods for the construction of test statistics does not apply. However, in all of the image-embedded datasets we have studied thus far, the behavior of test statistics and the selection methods has not differed noticeably from what would be expected under usual linear model assumptions.

7. ADDED-VARIABLE PLOTS

The computational algorithms can be used in ways other than for embedding images in the plot of residuals versus fitted values. Suppose that $\mathbf{R}_0$ and $\mathbf{T}$ are such that $\mathbf{R}_0^T\mathbf{T} = 0$, $\mathbf{R}_0^T\mathbf{1}_n = 0$ and the scatterplot of $\mathbf{R}_0$ versus $\mathbf{T}$ displays an image or message. Let $\mathbf{T}^*$ be a random permutation of the rows of $\mathbf{T}$. Next, set $\widehat{\mathbf{Y}}_0$ equal to the residual vector from the least squares regression of $\mathbf{T}^*$ on $[\mathbf{1}_n : \mathbf{R}_0]$. Now apply the data construction algorithm in Section 2 to the orthogonal residual and prediction vectors $\mathbf{R}_0$ and $\widehat{\mathbf{Y}}_0$, resulting in data $(\mathbf{X}, \mathbf{Y})$. Then the augmented data $([\mathbf{X} : \mathbf{T}], \mathbf{Y})$ is such that the "correct" model includes only the variables in $\mathbf{X}$. This is because $\mathbf{T}$ is orthogonal to the residuals from the regression of $\mathbf{Y}$ on $[\mathbf{1}_n, \mathbf{X}]$, and the $\mathbf{X}$ matrix is constructed independently of $\mathbf{T}$ ($\mathbf{T}$ is not orthogonal to the columns of $\mathbf{X}$, but they will be nearly so).

Thus, with a large dataset the coefficient of $\mathbf{T}$ in the full model that includes $\mathbf{X}$ and $\mathbf{T}$ will generally be very near zero and nonstatistically significant (p value ≈ 1). Thus any sensible variable selection approach will discard $\mathbf{T}$ and keep all the variables in $\mathbf{X}$. In addition, the residual and prediction vectors from the correct model are $\mathbf{R}_0$ and $\widehat{\mathbf{Y}}_0$ and the scatterplot of $\mathbf{R}_0$ versus $\widehat{\mathbf{Y}}_0$ is essentially patternless. However, the residual plot of $\mathbf{R}_0$ versus $\mathbf{T}$ manifests the desired image or message. Note that this is not the added-variable (or partial regression) plot $\mathbf{R}_0$ versus $(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{T}$. However, because of the near orthogonality of $\mathbf{T}$ and the columns of $\mathbf{X}$ the two plots are generally similar.

Figure 13 displays plots of $\mathbf{R}_0$ versus $\mathbf{T}$ and $\mathbf{R}_0$ versus $(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{T}$ for three datasets constructed in the manner described above. In the first dataset the ASA logo is embedded in the plot. The second dataset is constructed to be consistent with a situation in which $\mathbf{T}$ does not enter linearly into the regression model for $\mathbf{Y}$, but the added-variable plot indicates including it as a quadratic. The third dataset is constructed to mimic a case in which $\mathbf{T}$ does not enter linearly, but there is evidence that it should enter via an appropriate sinusoidal transformation.

8. SUMMARY

We provided an algorithm for generating multiple linear regression data having fixed residuals and predicted values, and shown how to exploit the algorithm to embed hidden images and messages in residual plots and added-variable plots. The method is useful for constructing interesting multiple linear regression datasets for classroom examples and exercises.

For most undergraduate regression courses, the primary appeal of the method is the construction of interesting and amusing datasets for simple model-building exercises. For graduate-level regression courses, particularly ones taught to students familiar with linear models theory at the level of say Christensen (2002), the method provides not only amusing examples, but also a means of generating curiosity in the method's workings to the extent that inquisitive students will reinforce their understanding of linear models theory as they try to master the details of the algorithm.

By the time this article appears in print the author's Web page will contain a link to multiple versions (different p, R_0^2) of the datasets whose residual plots appear in this article. Thus, instructors who only want to use the datasets for exercises can download them. In addition, the Web page will contain variable-selection *challenge* datasets of various levels of difficulty in which the so-called correct model will not be revealed in advance, but it will be evident by its residual plot. For instructors who want to embed their own images or messages in datasets, the Web page will also contain GAUSS and R programs for implementing all of the computational aspects of the algorithms described in this article. However, images usually require some preprocessing as described in the Appendix.

APPENDIX: COMPUTING, MESSAGE AND IMAGE PLOT CONSTRUCTION

The author's Web page contains GAUSS and R programs for implementing all facets of the dataset construction computations described in this article starting with a not-necessarily-orthogonal $\mathbf{R}_0$ and $\widehat{\mathbf{Y}}_0$. In addition there is a program that will convert text strings to scatterplots for embedding messages in residual plots. However, when working with images there are some nonstatistical computations that are required to convert an image to a scatterplot. We now summarize the main steps

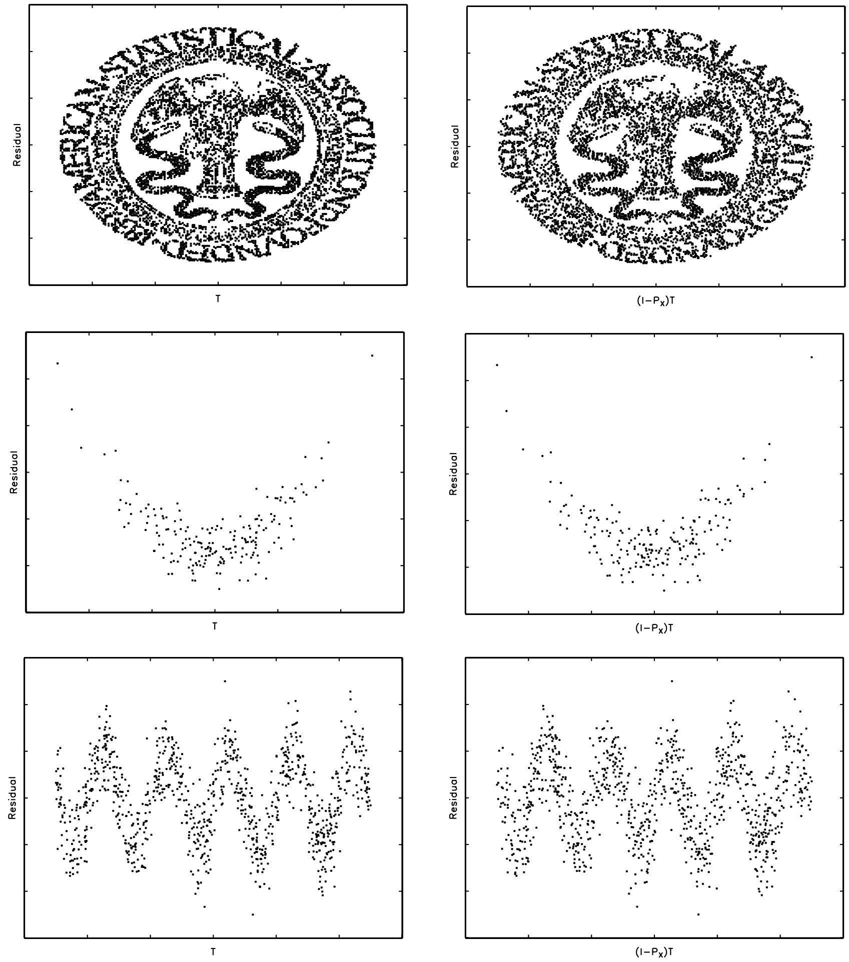


Figure 13. Added-variable plots. Three sets of added-variable plots. Left column, plots of $\mathbf{R}_0$ versus $\mathbf{T}$; right column, plots of $\mathbf{R}_0$ versus $(\mathbf{I}_n - \mathbf{P}_{\mathbf{X}_*})\mathbf{T}$. Top row, embedded image example; middle row, quadratic dependence; bottom row, sinusoidal dependence.

needed to take an image and convert it into a scatterplot that can be embedded into a residual plot.

Starting with a black-and-white image, and a black-and-white line drawing in particular is easiest, but not necessary. Readily available image-editing programs generally have the capabilities of converting any image to black-and-white. However, a simple conversion often does not result in a useful (for our purposes) black-and-white image. In order to keep file size reasonable the objective is to minimize the number of black pixels while maintaining recognizability of the image. Edge detection, or edge enhancement tools, also common in image-editing software, are useful for this purpose. In addition to using software tools, a little manual editing also might be necessary. An edge detection tool worked well on the image of Gertrude Cox in Figure 6 (p. 169). However, the image of Gauss in Figure 5 (p. 169) required much manual editing, primarily to remove numerous black pixels in his hat leaving only an outline of the hat. The bottom line is that a little digital artwork is required to transform an image to an acceptable black-and-white image.

The next step is to convert the black-and-white image to a collection of abscissa-ordinate pairs. For this conversion we used the image conversion utility “convert” in the public domain image-editing software, ImageMagick 6.3.0 to convert an image (e.g., jpg, gif, etc.) to a text file. Then one of the GAUSS or R programs on the author’s Web page reads and parses the image text file and outputs (x, y) pairs (i.e., a simple two-column dataset) of the black pixel locations. This output dataset is the nonorthogonal residuals and predicted values, $\mathbf{R}_0$ and $\widehat{\mathbf{Y}}_0$, assumed in Section 2.

IMAGE SOURCES

The image of Homer Simpson is a modified version of an original downloaded from *http://www.mathsci.appstate.edu/~sjg/simpsonsmath/blackboard.html*.

The image of the bison is a modified version of an original downloaded from *http://www.kidsplanet.org/tt/elemenlessons/bison.html*.

The image of R. A. Fisher is a modified version of an original downloaded from *http://www.csse.monash.edu.au/~lloyd/tildeImages/People/Fisher.RA/*.

The image of C. F. Gauss is a modified version of an original downloaded from *http://www.maths-rometus.org/mathematiques/histoire-des-maths/mathematicien/carl-friedrich-gauss.asp*

The image of G. Cox is a modified version of an original downloaded from *http://www-history.mcs.st-andrews.ac.uk/~history/PictDisplay/Cox.html*.

The image of T. Bayes is a modified version of an original downloaded from *http://www.lps.uci.edu/~johnsonk/CLASSES/inductivelogic/inductivelogicmain.html*.

The image of three trout is a modified version of an original downloaded from *http://www.sherpaguides.com/tennessee/great_smoky_mtns_np/fisheries.html*

The image of the lone trout is a modified version of an original downloaded from *http://www.megavision.net/ornament/wildlife_collection.htm*.

The image of the ASA logo is a modified version of an original downloaded from *http://www.amstat.org/about/index.cfm?fuseaction=main*.

[Received January 2007. Revised February 2007.]

REFERENCES

Christensen, R. (2002), *Plane Answers to Complex Questions* (3rd ed.) New York: Springer Verlag.