TOPICS IN STATISTICAL COMPUTING

Rolling Your Own: Linear Model Hypothesis Testing and Power Calculations via the Singular Value Decomposition

Steve Verrill, Mathematical Statistician,
USDA Forest Service Forest, Madison
steve@wsl3.fpl.fs.fed.us

Abstract

We outline the steps that would permit a statistician to produce special purpose linear model routines through the use of high quality public domain numerical analysis software.

Introduction

Good commercial linear model packages are readily available. It sometimes happens, however, that one would like linear model code that could be incorporated into a simulation. Verrill (1999) discusses such a simulation in the context of predictor sort sampling. See http://ws13.fpl.fs.fed.us/ttconf.html for a web interface to the simulation program.

Further, a sophisticated user can sometimes become frustrated with the inflexibility of a commercial package. This can be particularly true if the user is confronted with unbalanced data or complex hypotheses. In addition, some commercial linear models packages do not include the ability to perform power calculations.

In such cases the user can make use of public domain computer routines that yield flexible linear model capabilities. In this note we step potential users through the computations needed to perform hypothesis tests and power calculations. We follow the theoretical approach of Scheffe (1959). To do the numerical work we make use of the singular value decomposition (see, for example, Thisted (1988)). There are, of course, other numerical techniques that can be used to perform the necessary calculations (see, for example, Kennedy and Gentle (1980), Gentle (1998)). We focus on the singular value decomposition because it yields an approach that is numerically stable, reasonably efficient, and simple to explain and implement. We also suggest the use of the DCDFLIB public domain package of distribution routines.

The relation between the singular value decomposition, least squares, generalized inverses, and estimability has been discussed in Good (1969) and Eubank and Webster (1985).

Hypothesis Testing

The standard linear model is

$$\mathbf{y} = \mathbf{X}\beta + \epsilon,$$

where y is the n x 1 vector of responses, $\mathbf{X}$ is the $n \times p$ design matrix, ß is the $p \times 1$ parameter vector, and E is the $n \times 1$ vector of random errors. We assume that the $\mathbf{e}_i$ are independent, identically distributed $N(0, s^2)$ random variables.

We want to test a hypothesis of the form

$$\begin{aligned} \mathbf{c}_1^T\beta &= \eta_1 \\ \mathbf{c}_2^T\beta &= \eta_2 \\ &\vdots \\ \mathbf{c}_q^T\beta &= \eta_q. \end{aligned} \qquad (1)$$

(It is often the case that $c_i^T 1 = 0$, in which case c_i^T is referred to as a "contrast.")

For example, in a one-way ANOVA we are testing

$$\begin{aligned} (\,1 \;\; -1 \;\; 0 \;\; 0 \;\; \ldots \;\; 0\,)\beta &= 0 \\ (\,1 \;\; 0 \;\; -1 \;\; 0 \;\; \ldots \;\; 0\,)\beta &= 0 \\ &\vdots \\ (\,1 \;\; 0 \;\; 0 \;\; \ldots \;\; 0 \;\; -1\,)\beta &= 0. \end{aligned}$$

To test hypothesis (1) we proceed in a series of steps:

Is the Hypothesis Overspecified?

We need to determine whether the c_i are linearly independent. (If they are not, the hypothesis is overspecified. The user needs to think more clearly about the hypothesis, and arrive at a set of c_i's that *are* linearly independent.)

Let

$$\mathbf{C}_{p\times q} = (\mathbf{c}_1 \ldots \mathbf{c}_q)$$

where $q \le p$. The singular value decomposition of C is

$$\begin{aligned}\mathbf{C} &= \mathbf{U}_{p\times q}\begin{pmatrix}\gamma_1 & 0 & \ldots & 0 & 0\\ 0 & \gamma_2 & 0 & \ldots & 0\\ & & \vdots & & \\ 0 & 0 & \ldots & 0 & \gamma_q\end{pmatrix}\mathbf{V}^T_{q\times q}\\ &= \mathbf{U}\mathbf{D}_\gamma\mathbf{V}^T,\end{aligned}$$

where the columns of **U** are orthonormal to each other, **V** is an orthogonal matrix, and $\mathbf{D}_g$ is the diagonal matrix with $g = g_1 \ge g_2 \ge \ldots \ge g_q \ge 0$, the singular values of **C,** as its diagonal. Thus the rank of **C** is just the number of nonzero singular values. Because of the limitations of computer arithmetic, the null g_i's will not in general be exactly equal to zero. We need to determine a threshold value. If a g_i lies below that threshold, we take it to be equal to zero and conclude that the hypothesis is overspecified. A threshold value that is suggested in the numerical analysis literature (for example, Golub and Van Loan (1996)) is

$$\|\mathrm{C}\| \times (\text{the machine precision}).$$

Our experience suggests that

$$\text{threshold} = \|\mathrm{C}\| \times (\text{the machine precision}) \times 10$$

is a better rule of thumb. Recall that

$$\begin{aligned}\|\mathbf{C}\| &= \sqrt{\sum_{i=1}^{p}\sum_{j=1}^{q} c_{ij}^2} = \sqrt{\mathrm{trace}(\mathbf{C}\mathbf{C}^T)}\\ &= \sqrt{\mathrm{trace}(\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}\mathbf{U}^T)}\\ &= \sqrt{\mathrm{trace}(\mathbf{U}\mathbf{D}^2\mathbf{U}^T)}\\ &= \sqrt{\mathrm{trace}(\mathbf{D}^2\mathbf{U}^T\mathbf{U})} = \sqrt{\gamma_1^2 + \ldots + \gamma_q^2}.\end{aligned}$$

For double precision arithmetic on 32 bit computers one would use

$$\sqrt{\gamma_1^2 + \ldots + \gamma_q^2} \times 10^{-15}$$

as the threshold value.

Are the c_i's Estimable?

We need to know whether there exists an a_i that satisfies

$$E(\mathbf{a}_i^T\mathbf{y}) = \mathbf{c}_i^T\beta$$

for all ß where *E* is the expectation operator. This holds true if and only if there exists an a_i that satisfies

$$\mathbf{a}_i^T\mathbf{X} = \mathbf{c}_i^T \text{ or } \mathbf{c}_i = \mathbf{X}^T\mathbf{a}_i. \tag{2}$$

We determine whether equation (2) holds by first obtaining a singular value decomposition of the design matrix, **X.**

$$\mathbf{X}_{n\times p} = \mathbf{U}_{n\times r}\Lambda\mathbf{V}^T_{r\times p} \tag{3}$$

where the l 's are nonzero ($r \le p$ is the rank of **X**), or

$$\mathbf{X}^T = \mathbf{V}\Lambda\mathbf{U}^T.$$

Thus the projection operator onto the space spanned by the columns of X^T, $L(X^T)$, is VV^T, and $c_i = \mathbf{X}^T a_i$ for some a, if and only if c, Î $L(X^T)$ or $VV^Tc_i = c_i$. Using double precision arithmetic on 32 bit machines, one would take them to be equal if

$$\|\mathbf{V}\mathbf{V}^T\mathbf{c}_i - \mathbf{c}_i\| \le \|\mathbf{c}_i\| \times 10^{-15}.$$

Find the a_i that Satisfies $\mathbf{a}_i^T\mathbf{X} = \mathbf{c}_i^T$

Given that c_i is estimable, there is a unique a_i in the linear span of the columns of X, $L(\mathbf{X})$, such that

$$\mathbf{a}_i^T\mathbf{X} = \mathbf{c}_i^T.$$

(This is one of Scheffé's lemmas.)

Using the singular value decomposition of X, equation (3), we know that the solution of $c_i = X^Ta_i$ satisfies

$$\mathbf{c}_i = \mathbf{V}\Lambda\mathbf{U}^T\mathbf{a}_i$$

or

$$\mathbf{U}\Lambda^{-1}\mathbf{V}^T\mathbf{c}_i = \mathbf{U}\mathbf{U}^T\mathbf{a}_i. \tag{4}$$

But because a, Î $L(\mathbf{X})$ and UU^T is the projection operator onto $L(\mathbf{X})$, from equation (4) we have

$$\mathbf{U}\Lambda^{-1}\mathbf{V}^T\mathbf{c}_i = \mathbf{U}\mathbf{U}^T\mathbf{a}_i = \mathbf{a}_i.$$

Thus, we can use the left-hand side of the equation above to calculate the a_i that satisfies $\mathbf{a}_i^T\mathbf{X} = \mathbf{c}_i^T$.

Find an Orthonormal Basis for the Linear span of $\mathbf{a}_1, \ldots, \mathbf{a}_q$ — $L(\mathbf{A})$

We simply perform a singular value decomposition of $A = (a_1 \ldots a_q)$ to obtain

$$\mathbf{A}_{n\times q} = \mathbf{U}_{n\times q}\Delta\mathbf{V}^T_{q\times q}$$

and the columns of U form an orthonormal basis for $L(\mathbf{A})$.

Find the Hypothesis Sum of Squares

Case 1 —$h_1 = \ldots = h_q = 0$ Let $u_{A,1}, \ldots, u_{A,q}$ be an othonormal basis for $L(\mathbf{A})$ found as described in the preceding subsection. Scheffé's theory tells us that the hypothesis sum of squares is

$$HSS \equiv \sum_{i=1}^{q}\left(\mathbf{u}_{A,i}^T\mathbf{y}\right)^2$$

with q degrees of freedom.

Case 2 — at least one of the h's is nonzero

Scheffé's theory tells us that the hypothesis sum of squares is

$$HSS \equiv \left(\mathbf{A}^T\mathbf{y} - \eta\right)^T \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \left(\mathbf{A}^T\mathbf{y} - \eta\right).$$

with q degrees of freedom.

We can make use of the singular value decomposition of $\mathbf{A}$ given above to obtain

$$\left(\mathbf{A}^T\mathbf{A}\right)^{-1} = \mathbf{V}\Delta^{-2}\mathbf{V}^T. \quad (5)$$

Then taking

$$\mathbf{w} \equiv \mathbf{V}^T\left(\mathbf{A}^T\mathbf{y} - \eta\right),$$

we have

$$HSS = \sum_{i=1}^{q} w_i^2/\delta_i^2.$$

Find the Residual Sum of Squares

Let $\mathbf{u}_{X,1}, \ldots, u_{X,r}$ be the columns of the U matrix of the singular value decomposition, equation **(3),** of **X.** Then the projection of y onto the linear span of the columns of X, $P_{L(x)}$ (y), equals

$$\left((\mathbf{u}_{X,1})^T\mathbf{y}\right)\mathbf{u}_{X,1} + \ldots + \left((\mathbf{u}_{X,r})^T\mathbf{y}\right)\mathbf{u}_{X,r}$$

and the residual sum of squares equals

$$\begin{aligned} RSS &\equiv \left(\mathbf{y} - P_{L(\mathbf{X})}(\mathbf{y})\right)^T\left(\mathbf{y} - P_{L(\mathbf{X})}(\mathbf{y})\right) \\ &= \mathbf{y}^T\mathbf{y} - ((\mathbf{u}_{X,1})^T\mathbf{y})^2 - \ldots - ((\mathbf{u}_{X,r})^T\mathbf{y})^2 \end{aligned}$$

with n - r degrees of freedom.

Form the F Statistic and Compare It to the Appropriate Critical Values

The F statistic equals

$$(HSS/q)\,/\,(RSS/(n-r)).$$

This should be compared to an $F_{q,n-r}(1 - a)$ critical value where a is the significance level.

Confidence Intervals

Suppose that we are interested in confidence intervals on l estimable combinations of the parameters, $\mathbf{c}_1^T\beta$, $\ldots, \mathbf{c}_l^T\beta$, and further suppose that the linear span of the c's has rank $q \leq l$. This rank can be determined as described in section 2.1. Let $s^2 \equiv RSS/(n - r)$. Then (see Scheffé (1959)) we know that an individual (1 - a) x 100% confidence interval for $\mathbf{c}_j^T\beta$ is

$$\mathbf{a}_j^T\mathbf{y} \pm s \times t_{n-r}(\alpha/2) \times \sqrt{\mathbf{a}_j^T\mathbf{a}_j}$$

where $t_{n-r}(a/2)$ is the appropriate t critical value, and a_j satisfies $\mathbf{a}_j^T\mathbf{X} = \mathbf{c}_j^T$ (see section 2.3). Also, joint $(1 - a) \times 100\%$ confidence intervals for the $\mathbf{c}_j^T\beta$'s, $j \in \{1,\ldots,l\}$, are given by

$$\mathbf{a}_j^T\mathbf{y} \pm s\sqrt{q \times F_{q,n-r}(1-\alpha) \times \mathbf{a}_j^T\mathbf{a}_j}$$

where $F_{q,n-r}(1 - a)$ is the appropriate F critical value.

Power Calculations

The noncentrality parameter is given by

$$\begin{aligned} NCP &\equiv \left(\mathbf{A}^T\mathbf{X}\beta - \eta\right)^T\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\left(\mathbf{A}^T\mathbf{X}\beta - \eta\right)/\sigma^2 \\ &= \left(\mathbf{C}^T\beta - \eta\right)^T\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\left(\mathbf{C}^T\beta - \eta\right)/\sigma^2. \quad (6) \end{aligned}$$

Note that $(A^TA)^{-1}$ can be calculated as in equation (5).

Scheffé's noncentrality parameter equals the square root of our noncentrality parameter. Our version corresponds with what the DCDFLIB library (see below) expects.

In a completely general power calculation program, possible values of ß, h, and s^2 would be specified by a user in the course of power calculations. In the common case in which h = 0 and the ß's represent treatment means, it might be more reasonable to expect a user to specify the components of the ß vector as fractions of an overall mean and to specify a range of coefficients of variation. This would yield b/s values that would enable one to calculate *NCP*.

Under the null hypothesis, $\mathbf{C}^T\beta - \eta = \mathbf{0}$ and the noncentrality parameter is 0, but under the alternative hypothesis $C^T/ß \neq h$ and the noncentrality parameter is inflated above zero. If we operate at an a significance level, the power is given by

$$\text{Power} = \text{Prob}\left(F_{q,n-r,NCP} > F_{q,n-r}(1-\alpha)\right), \quad (7)$$

the probability that a noncentral F distribution with q numerator degrees of freedom, n - r denominator degrees of freedom, and noncentrality parameter *NCP* lies above the 100(1 - a)th percentile of a central F distribution with q numerator degrees of freedom and n - r denominator degrees of freedom.

implementing the theory as code

Public domain FORTRAN or C code to perform the singular value decomposition is found in the LINPACK (Dongarra and others (1979)) or (C)LAPACK (Anderson and others (1995)) linear algebra libraries. These can be obtained over the internet at http://www.netlib.org/liblist.html. Public domain C++ code to perform the singular value decomposition can be found by searching on svd at

http://www.netlib.org. A Java translation of the LINPACK singular value decomposition can be found at http://www1.fpl.fs.fed.us/linear_algebra.html. This site also points to Java translations of the LAPACK linear algebra routines.

Public domain FORTRAN or C code to calculate the *t* distribution and the central and noncentral *F* distributions and their inverses can be found in the DCDFLIB library. DCDFLIB is a public domain library of "routines for cumulative distribution functions, their inverses, and their parameters." It was produced by Barry Brown, James Lovato, and Kathy Russell of the Department of Biomathematics, M.D. Anderson Cancer Center, The University of Texas. DCDFLIB can be found at http://odin.mdacc.tmc.edu/anonftp/page_2.html.

A relatively raw example of the use of the LINPACK and DCDFLIB routines to create a linear models program can be obtained (or run over the Web) at http://www1.fpl.fs.fed.us/glm.html. This site includes sample input and output based on Table 9.1 in Milliken and Johnson (1992). We note that linear model routines produced by these methods would need extra work to become user friendly. In particular we have finessed the issue af the generation or input of the design, **X,** and hypotheses, C^T, matrices. We would expect a sophisticated user interested in simulations or special purpose analyses to be able to generate these matrices by hand. A person new to this approach might want to take a look at some of the examples in Milliken and Johnson (1992).

An algorithm that automatically generates design matrices for balanced factorial experiments is described in MacKenzie and O'Flaherty (1982).

Kennedy and Gentle (1980) (page 388) note:

> [User friendly] computer software must include the ability to accept user specification of the model. Most programs in use today allow the user to provide some rather natural algebraic specification. The program then deciphers the specification and translates it into numeric coding for subsequent use. There are no established standards for doing this, but many techniques used in compiler construction can be applied to this problem.

An example of such an algebraic specification is discussed in Wilkinson and Rogers (1973).

References

Anderson, E., Bai, Z., Bischof, C., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Ostrouchov, S., and Sorensen, D. (1995). *LAPACK Users' Guide, Second Edition.* Society of Industrial and Applied Mathematics, Philadelphia.

Dongarra, J., Moler, C., Bunch, J., and Stewart, G. (1979). *LINPACK Users' Guide.* Society of Industrial and Applied Mathematics, Philadelphia.

Eubank, R. L. and Webster, J. T. (1985). "The Singular-Value Decomposition as a Tool for Solving Estimability Problems," *The American Statistician*, **39**, 64–66.

Golub, Gene H. and Van Loan, Charles F. (1996). *Matrix Computations, Third Edition.* The Johns Hopkins University Press, Baltimore and London.

Gentle, J. E. (1998). *Numerical Linear Algebra for Applications in Statistics.* Springer-Verlag, New York.

Good, I. J. (1969). "Some Applications of the Singular Decomposition of a Matrix," *Technometrics*, **11**, 823–831.

Kennedy, W. J. and Gentle, J. E. (1980). Statistical Computing. Marcel Dekker, New York and Basel.

MacKenzie, G. and O'Flaherty, M. (1982). "Direct Design Matrix Generation for Balanced Factorial Experiments," *Applied Statistics*, **31**, 74–80.

Milliken, G. A. and Johnson, D. E. (1992). Analysis of Messy Data, Volume I: Designed Experiments. Chapman and Hall, London.

Scheffé, H. (1959). *The Analysis of Variance.* John Wiley and Sons, New York.

Thisted, R. A. (1988). *Elements of Statistical Computing.* Chapman and Hall, London.

Verrill, S. (1999). "When Good Confidence Intervals Go Bad: Predictor Sort Experiments and ANOVA," *The American Statistician*, **53**, 38–42.

Wilkinson, G. N. and Rogers, C. E. (1973). "Symbolic Description of Factorial Models for Analysis of Variance," *Applied Statistics*, **22**, 392–399.