Simplifying regression models using stepdown

Gareth Ambler
Department of Statistical Science, University College, 1-19 Torrington Place,
London WC1E 7HB, UK
g.ambler@ucl.ac.uk

Tony Brady
MRC Clinical Trials Unit, 222 Euston Road,
London NW1 2DA, UK
t.brady@ctu.mrc.ac.uk

9 May 2001

Introduction

`Stepdown` is a post-estimation command that approximates the prognostic index (linear predictor) of the previously fitted model to any given level of accuracy. That is, given a prognostic index (PI) $\mu = \beta_0 + \Sigma_j \beta_j x_j$, `stepdown` finds an approximate function $\mu^* = \beta_0^* + \Sigma_j \beta_j^* x_j$ where at least one $\beta_j^* = 0$. The closeness of μ^* to μ is determined by $R^2 = 1 - \Sigma(\mu - \mu^*)^2 / \Sigma(\mu - \overline{\mu})^2$. The R^2 statistic ranges from 1 ($\mu^* = \mu$) to 0 ($\mu^* = \overline{\mu}$) and may be obtained by regressing μ on μ^*. The stepdown idea was suggested by Harrell *et al* (1998).

Given a p parameter model, `stepdown` searches for the $(p-1)$ parameter model that maximises the R^2 criterion. At the next iteration it searches for the $(p-2)$ subset of the $(p-1)$ parameter model that maximises R^2. The algorithm continues until the model cannot be reduced in dimension with R^2 falling below some pre-determined value, `r2`. `r2` is zero by default but in practice users are more likely to want to choose a high value such as 0.95 or 0.99 to provide a good approximation to the PI.

As we will demonstrate in the examples, `stepdown` is particularly useful when we have a fitted a regression model with many predictors and wish to simplify it without using variable selection techniques such as forward selection or backwards elimination. This is especially appropriate when we are developing a clinical prediction rule and are using a full model. The latter uses as many predictors as possible with reference to the quantity of data (Harrell *et al*, 1996). By not using variable selection techniques the confidence intervals and P values from the full model are statistically valid. To increase its practical worth a more parsimonious model may be required since some less important predictors may be difficult or expensive to measure accurately. `Stepdown` therefore may be used to approximate the PI from the full model and produce a simplified PI with as many or few predictors as desired. The accuracy of the approximation is given by the value of R^2. The advantage of such an approach is that the responses are only used once to construct the initial PI and selection/omission bias is avoided since variable selection is not employed.

To demonstrate the routine we give examples using the well known `auto` dataset supplied with Stata and a breast cancer dataset previously analysed by Sauerbrei and Royston (1999).

The ado-file is a post estimation command with the following basic syntax:

`stepdown` [*term* ...] [`if` *exp*] [`in` *range*] [, *options*]

Details are given in the section *Syntax*.

Example 1: Automobile data

First we use regress to fit a model to predict `mpg` from `weight`, `length`, `displacement`, `price`, and country of origin (`foreign`) as follows:

```
. regress mpg weight length displacement price foreign

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  5,    68) =   28.06
       Model |  1645.78873     5  329.157747           Prob > F      =  0.0000
    Residual |  797.670726    68  11.7304519           R-squared     =  0.6735
-------------+------------------------------           Adj R-squared =  0.6495
       Total |  2443.45946    73  33.4720474           Root MSE      =   3.425

------------------------------------------------------------------------------
         mpg |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |  -.0042348   .0022018    -1.92   0.059    -.0086285    .0001589
      length |  -.0859926    .058598    -1.47   0.147    -.2029231    .0309379
displacement |   .0009477   .0102573     0.09   0.927    -.0195205    .0214158
       price |  -.0000407   .0002056    -0.20   0.844     -.000451    .0003695
     foreign |  -1.537816   1.360654    -1.13   0.262    -4.252959    1.177327
       _cons |   50.76626   6.431725     7.89   0.000     37.93196    63.60057
------------------------------------------------------------------------------
```

We note that `weight`, `length`, and `displacement` are highly correlated. We now use `stepdown` to simplify the prognostic index from this model using

```
. stepdown
drop in R2 = 0.000    removing displacement                         R2 = 1.000
drop in R2 = 0.000    removing price                                R2 = 1.000
drop in R2 = 0.016    removing length                               R2 = 0.984
drop in R2 = 0.017    removing foreign                              R2 = 0.967
drop in R2 = 0.967    removing weight                               R2 = 0.000
```

```
      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  0,    73) =    0.00
       Model |        0.00     0           .           Prob > F      =       .
    Residual |  1645.78874    73  22.5450513           R-squared     =  0.0000
-------------+------------------------------           Adj R-squared =  0.0000
       Total |  1645.78874    73  22.5450513           Root MSE      =  4.7482

------------------------------------------------------------------------------
          mu |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |    21.2973   .5519627    38.58   0.000     20.19724    22.39736
------------------------------------------------------------------------------
```

`stepdown` starts with the prognostic index (`predict mu, xb`) from the original regression model. Regressing `mu` on the 5 predictors produces a perfect fit with $R^2 = 1$. Next each predictor in turn is temporarily dropped from the model and `mu` is regressed on the remaining predictors. The omission of `displacement` causes the least decrease in R^2 so the predictor is permanently dropped. This procedure is continued until every predictor is dropped since the default value of the `r2` option is zero.

Although this leaves with an essentially useless model (simply the average of the original PI) it does summarise the effects of the predictors. We see that a PI including just `weight` would be a very good approximation to the original PI ($R^2 = 0.967$).

Example 2: Recurrence free survival time in breast cancer

This dataset consists of information on 686 patients with primary node positive breast cancer who were recruited by the German Breast Cancer Study Group (GBSG) between July 1984 and December 1989. Of these, 299 patients experienced at least one disease recurrence or died during the follow-up period. The median follow-up time was nearly 5 years. The data have been extensively analysed by Sauerbrei and Royston (1999), who used fractional polynomials to develop prognostic models. Here we consider Cox regression models for the relationship between recurrence-free survival time (`rectime`) with censoring variable `censrec` and the prognostic factors hormone therapy (`hormon`), age (`x1`), menopausal status (`x2`), tumour size (`x3`) and grade (`x4`), and the number of positive lymph nodes (`x5`), progesterone receptors (`x6`) and estrogen receptors (`x7`). `hormon` and `x2` are binary and `x4` is categorical with 3 levels (modelled using `x4a` and `x4b`).

We fit a cox regression model using second degree fractional polynomial (FP) terms (see `[r]` **fracpoly**) for `x1`, `x3`, `x6` and `x7` and an exponential term for `x5`; the latter provides a monotonic function which is consistent with existing medical knowledge. The form of the FP terms were determined using the ado-file `mfracpol` (Royston and Ambler 1999). We fit this model using

```
. fracgen x1 -2 -1
```

```
. fracgen x3 -0.5 0
. fracgen x6 0 3
. fracgen x7 2 2
. cox rectime hormon x1_* x2 x3_* x4a x4b x5e x6_* x7_*, dead(censrec)

Cox regression -- Breslow method for ties
Entry time 0                                      Number of obs   =        686
                                                  LR chi2(13)     =     161.59
                                                  Prob > chi2     =     0.0000
Log likelihood = -1707.3784                       Pseudo R2       =     0.0452

------------------------------------------------------------------------------
     rectime |
     censrec |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      hormon |  -.4051773   .1297879    -3.12   0.002    -.6595569   -.1507978
        x1_1 |   56.18261   13.82246     4.06   0.000     29.09108    83.27413
        x1_2 |  -22.98196   7.571521    -3.04   0.002    -37.82187   -8.142052
          x2 |   .1334753   .1949807     0.68   0.494      -.24868    .5156305
        x3_1 |  -.8678768   .7756834    -1.12   0.263    -2.388188    .6524347
        x3_2 |  -.6141592   .7547746    -0.81   0.416     -2.09349    .8651718
         x4a |   .4891838   .2519891     1.94   0.052    -.0047057    .9830734
         x4b |   .0093375   .1400539     0.07   0.947    -.2651631     .283838
         x5e |  -1.835175   .2417487    -7.59   0.000    -2.308994   -1.361356
        x6_1 |  -.1530925   .0356048    -4.30   0.000    -.2228767   -.0833084
        x6_2 |  -1.094637   1.009497    -1.08   0.278    -3.073215    .8839412
        x7_1 |   .8502001   .5086154     1.67   0.095    -.1466677    1.847068
        x7_2 |   1.066174   1.574662     0.68   0.498    -2.020107    4.152454
------------------------------------------------------------------------------
```

We now use stepdown to simplify the model to $R^2 = 0.9$. We wish to consider all linked FP terms together so we submit a varlist with linked terms entered within parenthesis. The other predictors do not need to specified since `stepdown` 'fills in' the varlist using the original estimates.

```
. stepdown (x1_*) (x3_*) (x4a x4b) (x6_*) (x7_*), r2(0.90)
drop in R2 = 0.002    removing x2                                    R2 = 0.998
drop in R2 = 0.007    removing x7_1 x7_2                             R2 = 0.992
drop in R2 = 0.024    removing x4a x4b                               R2 = 0.968
drop in R2 = 0.027    removing x3_1 x3_2                             R2 = 0.941
drop in R2 = 0.034    removing hormon                                R2 = 0.907

      Source |       SS       df       MS              Number of obs =     686
```

```
-------------+------------------------------           F(  5,   680) = 1327.06
       Model |  623.669491     5  124.733898           Prob > F      =  0.0000
    Residual |  63.9148313   680  .093992399           R-squared     =  0.9070
-------------+------------------------------           Adj R-squared =  0.9064
       Total |  687.584322   685  1.00377273           Root MSE      =  .30658

------------------------------------------------------------------------------
          mu |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        x1_1 |   57.43491   3.175662    18.09   0.000     51.19962    63.67019
        x1_2 |  -23.78261   1.481988   -16.05   0.000    -26.69243   -20.87279
        x6_1 |  -.1819911   .0062424   -29.15   0.000    -.1942478   -.1697344
        x6_2 |  -1.074799   .0210221   -51.13   0.000    -1.116075   -1.033523
         x5e |  -1.977133   .0484119   -40.84   0.000    -2.072188   -1.882079
       _cons |  -.7299701   .1714919    -4.26   0.000    -1.066687   -.3932528
------------------------------------------------------------------------------
```

We see that the `x2` and `x7` have virtually no effect on the prognostic index and that only `x1`, `x5` and `x6` are required to provide a reasonable approximation.

Syntax

`stepdown [`*term* `...]  [if` *exp*`] [in` *range*`] [,r2(#) lockterm1 exact hold trace nofill]`

where *term* is either a varname or (varlist) (a varlist in parentheses). Weights are not supported. `predict` after `stepdown` behaves the same as `predict` after `regress` (unless the `hold` option is used in which case it will behave as `predict` after the original estimation command.

Options

`r2(`#`)` determines the accuracy of the approximation where # lies between 0 and 1. The default of 0 drops all predictors.

`lockterm1` specifies that the first term is to be included in the model and not subjected to the selection criteria. By using parenthesis the first term can refer to a varlist.

`exact` specifies that an exact method is used. The default method (which is faster) makes use of the `test` command to calculate the drop in R^2 and does not fit as many regression models. However this method does not work when $R^2 = 1$ so a small amount of noise is added to the prognostic index to determine the first predictor to be omitted. The methods are identical thereafter.

`hold` keeps the original model estimates in the estimates list. This allows the user to repeat `stepdown` at a different level of accuracy. The default is for the final approximate model to occupy the estimates list.

`trace` gives more information about the various approximations considered.

`nofill` prevents stepdown from 'filling in' the varlist. This option is useful if the user wishes to specify a subset of the original varlist, and must be used if any new variables are specified.

Acknowledgement

The first author recieved financial support from project grant number 045512/Z/95/Z from the Wellcome Trust.

References

- Royston, P. and G. Ambler. 1999. sg81.1: Multivariable fractional polynomials: update. *Stata Technical Bulletin Reprints* vol 9, pp161-168.

- Sauerbrei, W. and P. Royston. 1999. Building multivariable prognostic and diagnostic models: transformation of the predictors using fractional polynomials. *Journal of the Royal Statistical Society (Series A)* 162: 71-94.

- Harrell, F. E., K. L. Lee, and D. B. Mark. 1996. Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine* 15: 361-387.

- Harrell, F. E., P. A. Margolis, S. Gove *et al.* 1998. Development of a clinical prediction model for an ordinal outcome. *Statistics in Medicine* 17: 909–944.