A program for saving a model fit as a data set

Roger Newson
Imperial College School of Medicine, London, UK
r.newson@ic.ac.uk

The command `parmest` is designed to save a model fit in a data set, either in memory, or on disk, or both. It was inspired by the example of `collapse`. It takes, as input, the parameter estimates of the most recently fitted model, and their covariance matrix. It creates, as output, a new data set, with one observation per parameter, and variables corresponding to equation names (if present), parameter names, estimates, standard errors, z- or t-test statistics, P-values and confidence limits. This output data set may be saved to a disk file, or remain in memory (overwriting the pre-existing data set), or both.

Typically, `parmest` is used with `graph` to produce confidence interval plots. It is also possible to sort the output data set by P-value, in order to carry out closed test procedures, like those of Holm, Hommel, or Holland and Copenhaver, summarized in Wright (1992).

Syntax

`parmest [,dof(#) label eform level(#) fast saving(`*filename*`[,replace]) norestore]`

Options

`dof(#)` specifies the degrees of freedom for t-distribution-based confidence limits. If `dof` is zero, then confidence limits are calculated using the standard normal distribution. If `dof` is absent, it is set to a default according to the last estimation results.

`label` indicates that a variable named label is to be generated in the new data set, containing the variable labels of variables corresponding to the parameter names, if such variables can be found in the existing data set.

`eform` indicates that the estimates and confidence limits are to be exponentiated, and the standard errors multiplied by the exponentiated estimates.

`level(#)` specifies the confidence level, in percent, for confidence limits. The default is `level(95)` or as set by `set level`. (See **[U] Estimation and post-estimation commands**.)

`fast` specifies that `parmest` not go to extra work so that it can restore the original data should the user press Break. `fast` is intended for use by programmers.

`saving(`*filename*`[,replace])` saves the output data set in a file. If `replace` is specified, and a file of name *filename* already exists, then the old file is overwritten.

`norestore` specifies whether or not the pre-existing data set is restored at the end of execution. This option is automatically set to `norestore` if `fast` is specified or `saving(`*filename*`)` is absent, otherwise it defaults to restoring the pre-existing data set.

Remarks

`parmest` creates a new data set with one observation per parameter and data on the most recent model fit. There are two character variables, `eq` and `parm`, containing equation and parameter names, respectively. The numeric variables are `estimate`, `stderr`, `z` (or `t`), `p`, `min`xx and `max`xx, where xx is the value of the `level` option. These variables contain parameter estimates, standard errors, z-test (or t-test) statistics, P-values, and confidence limits, respectively. The P-values test the hypothesis that the appropriate parameter is zero, or one if `eform` is specified.

Example

This example uses the Stata example data set `auto.dta`, with the added variable `manuf`, containing the first word of `make`, and denoting manufacturer. (See **[U] 26.10 Obtaining robust variance estimates** for an example of the use of this variable.) We want to derive confidence intervals for the average fuel efficiency (in miles per gallon) for each manufacturer, using a homoskedastic regression model. (Some manufacturers are represented by only one model in the data set, so their specific variances cannot be estimated.) We then want to plot the confidence intervals by manufacturer.

We proceed as follows. First we tabulate `manuf`, generating the dummy variables for the regression analysis:

```
. * CI plot *;
. tabulate manuf,missing gene(manu);
Manufacturer|      Freq.     Percent        Cum.
------------+-----------------------------------
        AMC |          3        4.05        4.05
       Audi |          2        2.70        6.76
        BMW |          1        1.35        8.11
      Buick |          7        9.46       17.57
       Cad. |          3        4.05       21.62
      Chev. |          6        8.11       29.73
     Datsun |          4        5.41       35.14
      Dodge |          4        5.41       40.54
       Fiat |          1        1.35       41.89
       Ford |          2        2.70       44.59
      Honda |          2        2.70       47.30
      Linc. |          3        4.05       51.35
      Mazda |          1        1.35       52.70
      Merc. |          6        8.11       60.81
       Olds |          7        9.46       70.27
    Peugeot |          1        1.35       71.62
      Plym. |          5        6.76       78.38
      Pont. |          6        8.11       86.49
    Renault |          1        1.35       87.84
     Subaru |          1        1.35       89.19
     Toyota |          3        4.05       93.24
         VW |          4        5.41       98.65
      Volvo |          1        1.35      100.00
------------+-----------------------------------
      Total |         74      100.00
```

We then carry out a regression analysis of `mpg` with respect to the dummy variables:

```
. regr mpg manu1-manu23,noconst;
  Source |       SS       df       MS                  Number of obs =      74
---------+------------------------------               F( 23,    51) =   70.51
   Model |  34910.1286    23  1517.83168               Prob > F      =  0.0000
Residual |  1097.87143    51  21.5268908               R-squared     =  0.9695
---------+------------------------------               Adj R-squared =  0.9558
   Total |    36008.00    74  486.594595               Root MSE      =  4.6397
------------------------------------------------------------------------------
     mpg |      Coef.   Std. Err.       t     P>|t|       [95% Conf. Interval]
---------+--------------------------------------------------------------------
   manu1 |   20.33333   2.678737      7.591   0.000       14.95555    25.71112
   manu2 |         20   3.280769      6.096   0.000       13.41358    26.58642
   manu3 |         25   4.639708      5.388   0.000        15.6854     34.3146
   manu4 |   19.14286   1.753645     10.916   0.000       15.62227    22.66345
   manu5 |   16.33333   2.678737      6.097   0.000       10.95555    21.71112
   manu6 |         22   1.894153     11.615   0.000       18.19733    25.80267
   manu7 |      25.75   2.319854     11.100   0.000        21.0927     30.4073
   manu8 |      20.25   2.319854      8.729   0.000        15.5927     24.9073
   manu9 |         21   4.639708      4.526   0.000        11.6854     30.3146
  manu10 |       24.5   3.280769      7.468   0.000       17.91358    31.08642
  manu11 |       26.5   3.280769      8.077   0.000       19.91358    33.08642
  manu12 |   12.66667   2.678737      4.729   0.000       7.288878    18.04445
  manu13 |         30   4.639708      6.466   0.000        20.6854     39.3146
  manu14 |   17.16667   1.894153      9.063   0.000          13.364    20.96934
  manu15 |   19.42857   1.753645     11.079   0.000       15.90798    22.94916
  manu16 |         14   4.639708      3.017   0.004       4.685398     23.3146
  manu17 |       26.2   2.074941     12.627   0.000       22.03438    30.36562
  manu18 |       19.5   1.894153     10.295   0.000       15.69733    23.30267
  manu19 |         26   4.639708      5.604   0.000        16.6854     35.3146
  manu20 |         35   4.639708      7.544   0.000        25.6854     44.3146
  manu21 |   22.33333   2.678737      8.337   0.000       16.95555    27.71112
  manu22 |       28.5   2.319854     12.285   0.000        23.8427     33.1573
  manu23 |         17   4.639708      3.664   0.001       7.685398     26.3146
------------------------------------------------------------------------------
```

We then use `parmest` to save the parameter estimates, and their confidence limits, to the new data set. This data set is `describe`d and `list`ed, to show the variables stored:

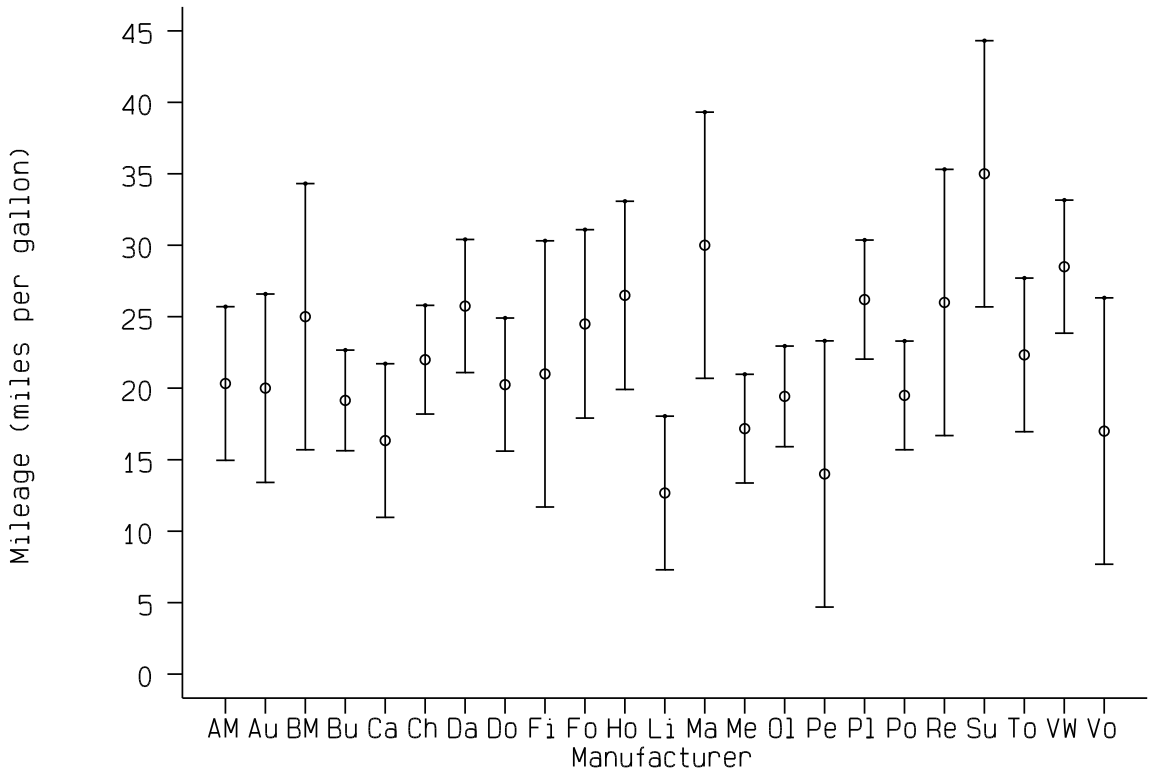
```
. parmest,lab;
. desc;
Contains data
  obs:            23
 vars:             8
 size:         1,656 (82.7% of memory free)
------------------------------------------------------------------------------
   1. parm         str6   %9s                 Parameter name
   2. label        str14  %14s                Parameter label
   3. estimate     double %10.0g              Parameter estimate
   4. stderr       double %10.0g              SE of parameter estimate
   5. t            double %10.0g              t-test statistic
   6. p            double %10.0g              P-value
   7. min95        double %10.0g              Lower 95% confidence limit
   8. max95        double %10.0g              Upper 95% confidence limit
------------------------------------------------------------------------------
Sorted by:
     Note:  data has changed since last save
```

```
. list parm label estimate stderr;
         parm           label    estimate      stderr
  1.     manu1      manuf==AMC   20.333333   2.6787367
  2.     manu2     manuf==Audi          20    3.280769
  3.     manu3      manuf==BMW          25    4.639708
  4.     manu4    manuf==Buick   19.142857   1.7536448
  5.     manu5     manuf==Cad.   16.333333   2.6787367
  6.     manu6    manuf==Chev.          22   1.8941529
  7.     manu7   manuf==Datsun       25.75    2.319854
  8.     manu8    manuf==Dodge       20.25    2.319854
  9.     manu9     manuf==Fiat          21    4.639708
 10.    manu10     manuf==Ford        24.5    3.280769
 11.    manu11    manuf==Honda        26.5    3.280769
 12.    manu12    manuf==Linc.   12.666667   2.6787367
 13.    manu13    manuf==Mazda          30    4.639708
 14.    manu14    manuf==Merc.   17.166667   1.8941529
 15.    manu15     manuf==Olds   19.428571   1.7536448
 16.    manu16  manuf==Peugeot          14    4.639708
 17.    manu17    manuf==Plym.        26.2   2.0749405
 18.    manu18    manuf==Pont.        19.5   1.8941529
 19.    manu19  manuf==Renault          26    4.639708
 20.    manu20   manuf==Subaru          35    4.639708
 21.    manu21   manuf==Toyota   22.333333   2.6787367
 22.    manu22       manuf==VW        28.5    2.319854
 23.    manu23    manuf==Volvo          17    4.639708
. list parm estimate min95 max95 t p;
         parm    estimate       min95       max95           t           p
  1.     manu1   20.333333   14.955545   25.711122   7.5906428   6.372e-10
  2.     manu2          20   13.413581   26.586419   6.0961317   1.450e-07
  3.     manu3          25   15.685398   34.314602   5.3882701   1.830e-06
  4.     manu4   19.142857   15.622268   22.663446    10.91604   5.972e-15
  5.     manu5   16.333333   10.955545   21.711122   6.0974016   1.443e-07
  6.     manu6          22    18.19733    25.80267   11.614691   6.151e-16
  7.     manu7       25.75   21.092699   30.407301   11.099836   3.265e-15
  8.     manu8       20.25   15.592699   24.907301   8.7289975   1.074e-11
  9.     manu9          21   11.685398   30.314602   4.5261469   .00003625
 10.    manu10        24.5   17.913581   31.086419   7.4677613   9.947e-10
 11.    manu11        26.5   19.913581   33.086419   8.0773745   1.100e-10
 12.    manu12   12.666667   7.2888785   18.044455   4.7285971   .00001823
 13.    manu13          30   20.685398   39.314602   6.4659241   3.796e-08
 14.    manu14   17.166667   13.363996   20.969337   9.0629784   3.306e-12
 15.    manu15   19.428571   15.907983    22.94916   11.078966   3.496e-15
 16.    manu16          14   4.6853976   23.314602   3.0174312   .00397119
 17.    manu17        26.2   22.034383   30.365617   12.626868   2.553e-17
 18.    manu18        19.5    15.69733    23.30267    10.29484   4.745e-14
 19.    manu19          26   16.685398   35.314602   5.6038009   8.504e-07
 20.    manu20          35   25.685398   44.314602   7.5435781   7.557e-10
 21.    manu21   22.333333   16.955545   27.711122   8.3372634   4.332e-11
 22.    manu22        28.5   23.842699   33.157301   12.285256   7.363e-17
 23.    manu23          17   7.6853976   26.314602   3.6640236   .00059118
```

We then augment this new data set with two new variables, the character variable `manufb` and the numeric variable `manufn`, derived from the variable labels stored in `label`, and representing the first two letters of the manufacturer's name. Finally, we use `manufn` to create a confidence interval plot for mean fuel efficiencies by manufacturer:

```
. gene str2 manufb=substr(label,length("manuf==")+1,2);
. encode manufb,gene(manufn);
```

Figure 1




```
. set textsize 100;
. grap estimate min95 max95 manufn,c(.II) s(O..)
>  xscale(0.5,23.5)
>  xlabel(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23)
>  yscale(0,45) ylabel(0,5,10,15,20,25,30,35,40,45)
>  t1title(" ") t2title(" ")
>  b2title("Manufacturer") l2title("Mileage (miles per gallon)")
>  saving(fig1.gph,replace);
```

The graph generated by this program is given as Figure 1.

Acknowledgements

I would like to thank Nick Cox of Durham University, UK, Jonah B. Gelbach at the University of Maryland at College Park, and Phil Ryan at the Department of Public Health, University of Adelaide, Australia for giving many helpful suggestions for improvements on previous versions posted to Statalist.

References

Wright SP. Adjusted P-values for Simultaneous Inference. *Biometrics* 1992; **48**: 1005-1013.