Title

stata.com

pkshape — Reshape (pharmacokinetic) Latin-square data

Syntax

```
pkshape id sequence period1 period2 [period list] [, options]
```

options	Description
`order(`*string*`)`	apply treatments in specified order
`outcome(`*newvar*`)`	name for outcome variable; default is `outcome(outcome)`
`treatment(`*newvar*`)`	name for treatment variable; default is `treatment(treat)`
`carryover(`*newvar*`)`	name for carryover variable; default is `carryover(carry)`
`sequence(`*newvar*`)`	name for sequence variable; default is `sequence(sequence)`
`period(`*newvar*`)`	name for period variable; default is `period(period)`

Menu

Statistics > Epidemiology and related > Other > Reshape pharmacokinetic latin-square data

Description

`pkshape` reshapes the data for use with `anova`, `pkcross`, and `pkequiv`; see [R] **anova**, [R] **pkcross**, and [R] **pkequiv**. Latin-square and crossover data are often organized in a manner that cannot be analyzed easily with Stata. `pkshape` reorganizes the data in memory for use in Stata.

`pkshape` is one of the pk commands. Please read [R] **pk** before reading this entry.

Options

`order(`*string*`)` specifies the order in which treatments were applied. If the `sequence()` specifier is a string variable that specifies the order, this option is not necessary. Otherwise, `order()` specifies how to generate the treatment and carryover variables. Any string variable can be used to specify the order. For crossover designs, any washout periods can be indicated with the number 0.

`outcome(`*newvar*`)` specifies the name for the outcome variable in the reorganized data. By default, `outcome(outcome)` is used.

`treatment(`*newvar*`)` specifies the name for the treatment variable in the reorganized data. By default, `treatment(treat)` is used.

`carryover(`*newvar*`)` specifies the name for the carryover variable in the reorganized data. By default, `carryover(carry)` is used.

`sequence(`*newvar*`)` specifies the name for the sequence variable in the reorganized data. By default, `sequence(sequence)` is used.

`period(`*newvar*`)` specifies the name for the period variable in the reorganized data. By default, `period(period)` is used.

Remarks and examples

stata.com

Often data from a Latin-square experiment are naturally organized in a manner that Stata cannot manage easily. `pkshape` reorganizes Latin-square data so that they can be used with `anova` (see [R] **anova**) or any pk command. This includes the classic 2×2 crossover design commonly used in pharmaceutical research, as well as many other Latin-square designs.

▷ Example 1

Consider the example data published in Chow and Liu (2009, 71). There are 24 patients, 12 in each sequence. Sequence 1 consists of the reference formulation followed by the test formulation; sequence 2 is the test formulation followed by the reference formulation. The measurements reported are the $\text{AUC}_{0-t_{\max}}$ for each patient and for each period.

```
. use http://www.stata-press.com/data/r13/chowliu
. list, sep(4)
```

	id	seq	period1	period2
1.	1	1	74.675	73.675
2.	4	1	96.4	93.25
3.	5	1	101.95	102.125
4.	6	1	79.05	69.45
5.	11	1	79.05	69.025
6.	12	1	85.95	68.7
7.	15	1	69.725	59.425
8.	16	1	86.275	76.125
9.	19	1	112.675	114.875
10.	20	1	99.525	116.25
11.	23	1	89.425	64.175
12.	24	1	55.175	74.575
13.	2	2	74.825	37.35
14.	3	2	86.875	51.925
15.	7	2	81.675	72.175
16.	8	2	92.7	77.5
17.	9	2	50.45	71.875
18.	10	2	66.125	94.025
19.	13	2	122.45	124.975
20.	14	2	99.075	85.225
21.	17	2	86.35	95.925
22.	18	2	49.925	67.1
23.	21	2	42.7	59.425
24.	22	2	91.725	114.05

Because the outcome for one person is in two different variables, the treatment that was applied to an individual is a function of the period and the sequence. To analyze this treatment using `anova`, all the outcomes must be in one variable, and each covariate must be in its own variable. To reorganize these data, use `pkshape`:

```
. pkshape id seq period1 period2, order(ab ba)
. sort seq id treat
```

```
. list, sep(8)
```

	id	sequence	outcome	treat	carry	period
1.	1	1	74.675	1	0	1
2.	1	1	73.675	2	1	2
3.	4	1	96.4	1	0	1
4.	4	1	93.25	2	1	2
5.	5	1	101.95	1	0	1
6.	5	1	102.125	2	1	2
7.	6	1	79.05	1	0	1
8.	6	1	69.45	2	1	2
9.	11	1	79.05	1	0	1
10.	11	1	69.025	2	1	2
11.	12	1	85.95	1	0	1
12.	12	1	68.7	2	1	2
13.	15	1	69.725	1	0	1
14.	15	1	59.425	2	1	2
15.	16	1	86.275	1	0	1
16.	16	1	76.125	2	1	2
17.	19	1	112.675	1	0	1
18.	19	1	114.875	2	1	2
19.	20	1	99.525	1	0	1
20.	20	1	116.25	2	1	2
21.	23	1	89.425	1	0	1
22.	23	1	64.175	2	1	2
23.	24	1	55.175	1	0	1
24.	24	1	74.575	2	1	2
25.	2	2	37.35	1	2	2
26.	2	2	74.825	2	0	1
27.	3	2	51.925	1	2	2
28.	3	2	86.875	2	0	1
29.	7	2	72.175	1	2	2
30.	7	2	81.675	2	0	1
31.	8	2	77.5	1	2	2
32.	8	2	92.7	2	0	1
33.	9	2	71.875	1	2	2
34.	9	2	50.45	2	0	1
35.	10	2	94.025	1	2	2
36.	10	2	66.125	2	0	1
37.	13	2	124.975	1	2	2
38.	13	2	122.45	2	0	1
39.	14	2	85.225	1	2	2
40.	14	2	99.075	2	0	1
41.	17	2	95.925	1	2	2
42.	17	2	86.35	2	0	1
43.	18	2	67.1	1	2	2
44.	18	2	49.925	2	0	1
45.	21	2	59.425	1	2	2
46.	21	2	42.7	2	0	1
47.	22	2	114.05	1	2	2
48.	22	2	91.725	2	0	1

Now the data are organized into separate variables that indicate each factor level for each of the covariates, so the data may be used with `anova` or `pkcross`; see [R] **anova** and [R] **pkcross**.

◁

Example 2

Consider the study of background music on bank teller productivity published in Kutner et al. (2005). The data are

Week	Monday	Tuesday	Wednesday	Thursday	Friday
1	18(D)	17(C)	14(A)	21(B)	17(E)
2	13(C)	34(B)	21(E)	16(A)	15(D)
3	7(A)	29(D)	32(B)	27(E)	13(C)
4	17(E)	13(A)	24(C)	31(D)	25(B)
5	21(B)	26(E)	26(D)	31(C)	7(A)

The numbers are the productivity scores, and the letters represent the treatment. We entered the data into Stata:

```
. use http://www.stata-press.com/data/r13/music, clear
. list
```

	id	seq	day1	day2	day3	day4	day5
1.	1	dcabe	18	17	14	21	17
2.	2	cbead	13	34	21	16	15
3.	3	adbec	7	29	32	27	13
4.	4	eacdb	17	13	24	31	25
5.	5	bedca	21	26	26	31	7

We reshape these data with `pkshape`:

```
. pkshape id seq day1 day2 day3 day4 day5
. list, sep(0)
```

	id	sequence	outcome	treat	carry	period
1.	3	1	7	1	0	1
2.	5	2	21	3	0	1
3.	2	3	13	5	0	1
4.	1	4	18	2	0	1
5.	4	5	17	4	0	1
6.	3	1	29	2	1	2
7.	5	2	26	4	3	2
8.	2	3	34	3	5	2
9.	1	4	17	5	2	2
10.	4	5	13	1	4	2
11.	3	1	32	3	2	3
12.	5	2	26	2	4	3
13.	2	3	21	4	3	3
14.	1	4	14	1	5	3
15.	4	5	24	5	1	3
16.	3	1	27	4	3	4
17.	5	2	31	5	2	4
18.	2	3	16	1	4	4
19.	1	4	21	3	1	4
20.	4	5	31	2	5	4
21.	3	1	13	5	4	5
22.	5	2	7	1	5	5
23.	2	3	15	2	1	5
24.	1	4	17	4	3	5
25.	4	5	25	3	2	5

Here the `sequence` variable is a string variable that specifies how the treatments were applied, so the `order` option is not used. When the sequence variable is a string and the `order` is specified, the arguments from the `order` option are used. We could now produce an ANOVA table:

```
. anova outcome seq period treat
```

```
                  Number of obs =      25     R-squared     =  0.8666
                  Root MSE      = 3.96232     Adj R-squared =  0.7331

          Source | Partial SS    df       MS           F     Prob > F
-----------------+----------------------------------------------------
           Model |      1223.6    12  101.966667       6.49     0.0014
                 |
        sequence |           82     4        20.5       1.31     0.3226
          period |        477.2     4        119.3       7.60     0.0027
           treat |        664.4     4        166.1      10.58     0.0007
                 |
        Residual |        188.4    12         15.7
-----------------+----------------------------------------------------
           Total |         1412    24  58.8333333
```

Example 3

Consider the Latin-square crossover example published in Kutner et al. (2005). The example is about apple sales given different methods for displaying apples.

Pattern	Store	Week 1	Week 2	Week 3
1	1	9(B)	12(C)	15(A)
	2	4(B)	12(C)	9(A)
2	1	12(A)	14(B)	3(C)
	2	13(A)	14(B)	3(C)
3	1	7(C)	18(A)	6(B)
	2	5(C)	20(A)	4(B)

We entered the data into Stata:

```
. use http://www.stata-press.com/data/r13/applesales, clear
. list, sep(2)
```

	id	seq	p1	p2	p3	square
1.	1	1	9	12	15	1
2.	2	1	4	12	9	2
3.	3	2	12	14	3	1
4.	4	2	13	14	3	2
5.	5	3	7	18	6	1
6.	6	3	5	20	4	2

Now the data can be reorganized using descriptive names for the outcome variables.

```
. pkshape id seq p1 p2 p3, order(bca abc cab) seq(pattern) period(order)
> treat(displays)
. anova outcome pattern order display id|pattern
```

```
                   Number of obs =      18     R-squared     =  0.9562
                   Root MSE      = 1.59426     Adj R-squared =  0.9069

          Source |  Partial SS    df       MS           F     Prob > F
-----------------+----------------------------------------------------
           Model |  443.666667     9  49.2962963       19.40     0.0002
                 |
         pattern |  .333333333     2  .166666667        0.07     0.9370
           order |  233.333333     2  116.666667       45.90     0.0000
        displays |         189     2        94.5       37.18     0.0001
      id|pattern |          21     3           7        2.75     0.1120
                 |
        Residual |  20.3333333     8  2.54166667
-----------------+----------------------------------------------------
           Total |         464    17  27.2941176
```

These are the same results reported by Kutner et al. (2005).

Example 4

We continue with example 1 from [R] **pkcollapse**; the data are

```
. use http://www.stata-press.com/data/r13/pkdata2, clear
. list, sep(4) abbrev(10)
```

	id	seq	auc_concA	auc_concB
1.	1	1	150.9643	218.5551
2.	2	1	146.7606	133.3201
3.	3	1	160.6548	126.0635
4.	4	1	157.8622	96.17461
5.	5	1	133.6957	188.9038
6.	7	1	160.639	223.6922
7.	8	1	131.2604	104.0139
8.	9	1	168.5186	237.8962
9.	10	2	137.0627	139.7382
10.	12	2	153.4038	202.3942
11.	13	2	163.4593	136.7848
12.	14	2	146.0462	104.5191
13.	15	2	158.1457	165.8654
14.	18	2	147.1977	139.235
15.	19	2	164.9988	166.2391
16.	20	2	145.3823	158.5146

```
. pkshape id seq auc_concA auc_concB, order(ab ba)
. sort period id
```

```
. list, sep(4)
```

	id	sequence	outcome	treat	carry	period
1.	1	1	150.9643	1	0	1
2.	2	1	146.7606	1	0	1
3.	3	1	160.6548	1	0	1
4.	4	1	157.8622	1	0	1
5.	5	1	133.6957	1	0	1
6.	7	1	160.639	1	0	1
7.	8	1	131.2604	1	0	1
8.	9	1	168.5186	1	0	1
9.	10	2	137.0627	2	0	1
10.	12	2	153.4038	2	0	1
11.	13	2	163.4593	2	0	1
12.	14	2	146.0462	2	0	1
13.	15	2	158.1457	2	0	1
14.	18	2	147.1977	2	0	1
15.	19	2	164.9988	2	0	1
16.	20	2	145.3823	2	0	1
17.	1	1	218.5551	2	1	2
18.	2	1	133.3201	2	1	2
19.	3	1	126.0635	2	1	2
20.	4	1	96.17461	2	1	2
21.	5	1	188.9038	2	1	2
22.	7	1	223.6922	2	1	2
23.	8	1	104.0139	2	1	2
24.	9	1	237.8962	2	1	2
25.	10	2	139.7382	1	2	2
26.	12	2	202.3942	1	2	2
27.	13	2	136.7848	1	2	2
28.	14	2	104.5191	1	2	2
29.	15	2	165.8654	1	2	2
30.	18	2	139.235	1	2	2
31.	19	2	166.2391	1	2	2
32.	20	2	158.5146	1	2	2

◁

We call the resulting dataset `pkdata3`. We conduct equivalence testing on the data in [R] **pkequiv**, and we fit an ANOVA model to these data in the third example of [R] **pkcross**.

References

Chow, S.-C., and J.-P. Liu. 2009. *Design and Analysis of Bioavailability and Bioequivalence Studies.* 3rd ed. Boca Raton, FL: Chapman & Hall/CRC.

Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. *Applied Linear Statistical Models.* 5th ed. New York: McGraw–Hill/Irwin.

Also see

[R] **pk** — Pharmacokinetic (biopharmaceutical) data