The Retain Statement

Programmer's Seminar
Feb. 07, 2008

The Retain Statement

- What is the Retain statement?
- Why use the Retain statement?
  - Reorder the dataset variables
  - Keeping counts or using counters
  - Recursive programming
  - Iterations
- How do we use Retain?

What is Retain?

- **Retain**
  - **(verb) to keep, hold, hold back, keep back**
- **Retain in SAS**
  - **RETAIN** *variable <=values> < ... V ariable <=values> >* **;**

Why use retain?

- Changing Variable Order
  - Large dataset contain useless variables that may contain important variables, however, might not be used till later in the future. Changing the order of variable output may save time and diminish error in the long run.

Retain – Code 1

```
data test.prescr2;
  retain pid start_date end_date
  drug_class supply_days;
  set junk.prescr2;
  keep pid supply_days start_date
  end_date drug_class;
run;
```

Data before Code 1

Obs	pid	supply_ days	drug_ class	start_date	end_date
1	1	10	CV100	12/13/2004	12/23/2004
2	1	10	CV400	12/13/2004	12/23/2004
3	1	90	CV100	12/23/2004	03/23/2005
4	1	90	CV400	12/23/2004	03/23/2005
5	1	90	CV800	02/08/2005	05/09/2005
6	1	90	CV100	03/22/2005	06/20/2005
7	1	30	CV200	03/22/2005	04/21/2005
8	1	90	CV100	03/23/2005	06/21/2005
9	1	90	CV800	04/29/2005	07/28/2005
10	1	30	CV200	05/13/2005	06/12/2005

Data After Code 1

Obs	pid	start_date	end_date	drug_ class	supply_ days
1	1	12/13/2004	12/23/2004	CV100	10
2	1	12/13/2004	12/23/2004	CV400	10
3	1	12/23/2004	03/23/2005	CV100	90
4	1	12/23/2004	03/23/2005	CV400	90
5	1	02/08/2005	05/09/2005	CV800	90
6	1	03/22/2005	06/20/2005	CV100	90
7	1	03/22/2005	04/21/2005	CV200	30
8	1	03/23/2005	06/21/2005	CV100	90
9	1	04/29/2005	07/28/2005	CV800	90
10	1	05/13/2005	06/12/2005	CV200	30

Why use retain?

- Working with Counters or Flags
  - Counters help as an alternative method of identifying data by observation. Where the first and last statement can point out the first and last entry to an observation, counters can identify every observation – first, last and in between.
  - Example – Pick out the 3rd prescription for each subject if possible.

Retain – Code 2

```
data junk.junk;
   set test.prescr2;
        by pid;
   retain x;
   if first.pid then x=1;
        else x=x+1;
run;
data junk.junk1;
   set junk.junk;
   if x=3;
run;
```

Data After Code 2 (first data step)

Obs	pid	start_date	end_date	drug_class	supply_days	x
1	1	12/13/2004	12/23/2004	CV100	10	1
2	1	12/13/2004	12/23/2004	CV400	10	2
3	1	12/23/2004	03/23/2005	CV100	90	3
4	1	12/23/2004	03/23/2005	CV400	90	4
5	1	02/08/2005	05/09/2005	CV800	90	5
6	1	03/22/2005	06/20/2005	CV100	90	6
7	1	03/22/2005	04/21/2005	CV200	30	7
8	1	03/23/2005	06/21/2005	CV100	90	8
9	1	04/29/2005	07/28/2005	CV800	90	9
10	1	05/13/2005	06/12/2005	CV200	30	10
11	1	06/13/2005	07/13/2005	CV200	30	11
12	1	07/28/2005	10/26/2005	CV800	90	12
13	1	08/02/2005	10/31/2005	CV200	90	13
14	1	09/09/2005	12/08/2005	CV100	90	14
15	2	05/24/2005	06/23/2005	CV200	30	1
16	2	07/06/2005	08/05/2005	CV200	30	2
17	2	11/23/2005	12/23/2005	CV200	30	3
18	2	12/13/2005	01/12/2006	CV200	30	4
19	2	01/22/2006	02/21/2006	CV200	30	5
20	2	02/24/2006	03/26/2006	CV200	30	6

Data After Code 2 (Second data step)

Obs	pid	start_date	end_date	drug_ class	supply_ days	x
1	1	12/23/2004	03/23/2005	CV100	90	3
2	2	11/23/2005	12/23/2005	CV200	30	3
3	3	04/19/2006	05/28/2006	CV800	39	3
4	4	01/19/2005	01/26/2005	Other	7	3
5	5	02/16/2005	03/03/2005	CV701	15	3
6	7	10/28/2004	11/27/2004	CV805	30	3
7	9	06/10/2005	09/08/2005	CV800	90	3
8	10	10/01/2004	12/30/2004	CV100	90	3
9	11	11/03/2005	12/18/2005	CV701	45	3
10	12	06/27/2005	08/29/2005	CV701	63	3

Something more Applicable

- Michael Sulik's Exrtreme Observation Program
  - Proc Univariate will provide the 5 highest and 5 lowest values – using retain and counts, we can provide x highest and y lowest values
  - See Extreme Observation Program

Alternatives to Procs

- The Retain statement can replace certain procedure or simplify certain procedures
  - Proc Univariate
  - Proc Freq
  - Proc Means (Sum)

Sample Data (junk.abc)

Obs	a	b	c
1	1	2	3
2	2	4	5
3	1	1	1
4	0	2	0
5	1	3	5
6	5	2	1
7	2	2	2

Running Total – Code 2a

```
data test.abc;
  set junk.abc;
  retain x 0 xa 2 xb xc 0;
  x = x + a + b + c;
  xa = xa + a;
  xb = xb + b;
  xc = xc + c;
run;
```

Sample Data after Datastep (test.abc)

Obs	a	b	c	x	xa	xb	xc
1	1	2	3	6	2	2	3
2	2	4	5	17	4	6	8
3	1	1	1	20	5	7	9
4	0	2	0	22	5	9	9
5	1	3	5	31	6	12	14
6	5	2	1	39	11	14	15
7	2	2	2	45	13	16	17

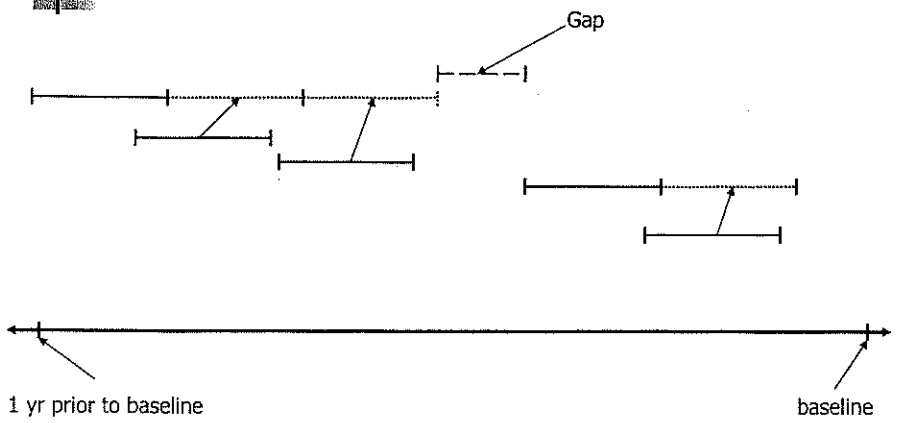
A Complex Example

- Medication Data
  - Medication Adherence
    - MPR = Total time on Drug / Total Time Prescribed
  - Gap Ratio
    - Gap Ratio = Total time off Drug / Total Time Prescribed

A Complex Example (cont'd.)

- Drug Prescriptions can not overlap
- Credit days must be added to the end of the overall prescriptions to encompass total drug time

Medications



Coding Brainstorm

- Lag time
  - Compare the time from the start of the next prescription with the end of the current prescription to check for overlap
- Macro Do-Loop
  - If overlap occurs, the new prescription time should be checked with the next prescription to check for overlap again
- Output statements
- If then do, else if then do, end.
- Retain simplifies the mess

Retain – Code 3

```
data a1;
    set prescr2;
        by pid;
    format sdate edate mmddyy10.;
    retain sdate edate;
    if first.pid then do;
        sdate=start_date;
        edate=end_date;
    end;
    else do;
        if edate>=start_date then do;
            edate=edate+(end_date-start_date);
        end;
        else do;
            sdate=start_date;
            edate=end_date;
        end;
    end;
run;
```

Data (pid 4)

Obs	pid	start_date	end_date	drug_ class	supply_ days
41	4	01/19/2005	01/26/2005	CV701	7
42	4	01/19/2005	01/26/2005	CV800	7
43	4	01/19/2005	01/26/2005	Other	7
44	4	04/14/2005	05/14/2005	CV701	30
45	4	04/19/2005	07/18/2005	CV701	90
46	4	04/19/2005	07/18/2005	CV800	90
47	4	04/19/2005	06/14/2005	Other	56
48	4	08/22/2005	11/20/2005	CV701	90
49	4	08/22/2005	11/20/2005	CV800	90
50	4	08/22/2005	10/17/2005	Other	56
51	4	12/16/2005	01/15/2006	CV100	30
52	4	12/16/2005	01/19/2006	CV701	34
53	4	12/16/2005	01/19/2006	CV800	34

Data cleaned up after Retain

Obs	pid	start_date	end_date	drug_class	supply_days	sdate	edate	gap_days
9	4	01/19/2005	01/26/2005	Other	21	01/19/2005	02/09/2005	0
10	4	12/16/2005	01/19/2006	CV800	600	04/14/2005	12/05/2006	64

Questions?

```
* Programmer: hopiy
* Date Created:  Feb. 04, 2008
* Date Edited: --
* Purpose: Create Gap ratio for (hypertesive) medication by any hypertensive example for
       programming seminar;

IBNAME TEST 'C:\TEST\TEST';
IBNAME JUNK 'C:\TEST\JUNK';
IBNAME BASE 'C:\TEST\BASE';
include 'R:\biostat\formats\formats.sas';

ata junk.prescr2;
   set prescr2;
   keep pid supply_days start_date end_date drug_class;
un;

**********************************************/
**          Retain statement begins          **/
**                   code 1                  **/
**********************************************/
ata test.prescr2;
   retain pid start_date end_date drug_class supply_days;
   set junk.prescr2;
un;

roc sort data=test.prescr2;by pid;run;

**********************************************/
**          Retain statement begins          **/
**                   code 2                  **/
**********************************************/
ata junk.junk;
   set test.prescr2;
       by pid;
   retain x;
   if first.pid then x=1;
       else x=x+1;
un;
ata junk.junk1;
   set junk.junk;
   if x=3;
un;

**********************************************/
**          Retain statement begins          **/
**                   code 2a                  **/
**********************************************/

ata junk.abc;
   input a b c;
ards;
 2 3
 4 5
 1 1
 2 0
```

```
 3 5
 2 1
 2 2

un;

ata test.abc;
   set junk.abc;
   retain x 0 xa 2 xb xc 0;
   x = x + a + b + c;
   xa = xa + a;
   xb = xb + b;
   xc = xc + c;
un;

**********************************************/
**          Retain statement begins          **/
**                   code 3                  **/
**********************************************/

*retain statment used to created combined drug prescriptions;
*different from jingsan's programming;
ata test.a1;
   set test.prescr2;
       by pid;
   format sdate edate mmddyy10.;
   retain sdate edate;
   if first.pid then do;
       sdate=start_date;
       edate=end_date;
   end;
   else do;
       if edate>=start_date then do;
           edate=edate+(end_date-start_date);
       end;
       else do;
           sdate=start_date;
           edate=end_date;
       end;
   end;
un;

roc sort data=a1;
   by pid sdate;
un;

*if last.sdate will only keep the combined prescriptions;
ata test.a2;
   set test.a1;
       by pid sdate;
   if last.sdate;
   supply_days=edate-sdate;
   gap_days=sdate-lag(edate);
un;
roc sort data=test.a2;
```

```
   by pid;
un;

ata test.a3;
   set test.a2;
       by pid;
   if first.pid then gap_days=.;
   if gap_days>0 then gap_any=1;
   if gap_days>=30 then gap_30=1;
   if gap_days>=60 then gap_60=1;
   if gap_days>=90 then gap_90=1;
   if gap_days=. then gap_days=0;
un;

roc univariate data=test.a3;
   var gap_days;
un;

roc means data=test.a3 noprint;
  by pid;
  var gap_days gap_any gap_30 gap_60 gap_90;
  output out=total_gap sum=total_gap total_gap_any total_gap_30 total_gap_60 total_gap_90;
un;

roc sort data=prescr2 out=days_on;by pid start_date;run;

ata days_on;
   set days_on;
       by pid start_date;
   if first.pid or last.pid;
   keep pid start_date;
un;

ata days_on;
   set days_on;
       by pid;
   start_date2=lag(start_date);
   if first.pid then delete;
   format start_date2 mmddyy10.;
   total_on=start_date-start_date2;
   rename start_date2=last_date;
un;

ata total_gap1;
   merge total_gap(in=a) days_on;
       by pid;
   if a;
   gap_ratio=total_gap/total_on;
un;

ata test.gap_ratio_anyhyper_def2;
   set total_gap1;
   if gap_ratio=. then gap_ratio=0;
un;
```