Training Session 1: Introduction to SAS

Rodney Sparapani, MS
Center for Patient Care and Outcomes Research
Medical College of Wisconsin

Biostatistics Research Seminar, MCW, May 27, 2009

Training Outline

1. **PCOR Hardware and Software**
2. **Brief History of UNIX**
3. **Brief History of Emacs**
4. **UNIX and Emacs Resources**
5. **PCOR Data Resource Examples**
6. **Brief History of SAS**
7. **SAS Training: Session 1**

If all else fails, read the instructions.
- Donald Knuth, renowned computer scientist

PCOR Hardware and Software

1. **UNIX Server (godzilla) and robotic backup system**
    - **2 dual-core CPUs (2.8 GHz)**
    - **16GB RAM**
    - **14TB disk space**
    - **server room access via fingerprint verification**
2. **Software Toolbox**
    - **Big guns: UNIX, SAS and Emacs/ESS (xemacs)**
    - **Secure access: ssh and X Server (X-Win32)**
    - **PCOR: MCWCORP Administrator access granted!**
    - **Little guns: R and Stata (2 users, no GUI)**
3. **Same username as your email address**
4. **Peer-to-Peer (P2P) file sharing coming RSN**

A Brief History of UNIX®

- **1969: AT&T Bell Labs starts work on UNIX**
- **1970: open source UNIX provided by AT&T for nominal fee, but no support (early shareware)**
- **1972-3: Bell Labs develops C, re-writes UNIX in C**
- **1973-8: DARPA invents TCP/IP**
- **1978: University of California releases Berkeley Software Distribution (BSD)**
- **1981-3: ARPANET goes TCP/IP (Internet)**
- **1987: MIT/DEC release the X Window System (X11)**
- **1990: AT&T merges UNIX and BSD (SVR4)**
- **1999-2000: OpenSSL/OpenSSH (BSD) are released**

A Brief History of Emacs and ESS

- **1975: Emacs created by Richard Stallman at MIT**
- **1984: Stallman creates GNU GPL whose goal is a**
  “complete, UNIX-compatible software system”
  re-writes GNU Emacs (GPL) in C
- **1990: Sall adds some SAS support to GNU Emacs**
- **1991: Lucid Emacs (GPL) for X11 released**
- **1994: GNU Emacs (GPL) for X11 released**
  Lucid defunct, SUN et al. rename it XEmacs
  Tom Cook releases SAS-mode (GPL)
- **1994-7: Anthony Rossini creates ESS (GPL) which contains ESS[SAS], ESS[S], ESS[Stata], etc.**
- **1999→: Rodney Sparapani and ESS team revamp ESS[SAS]**

UNIX and Emacs Resources

- **UNIX in a Nutshell by Daniel Gilly, O'Reilly**
- **Learning GNU Emacs by Cameron et al., O'Reilly**
- **GNU Emacs: UNIX Text Editing and Programming by Shoonover et al., Addison Wesley**

PCOR Data Resource Examples

1. **Geographic**
   - **Census (1990 and 2000)**
   - **Urban Intensity Code (2003)**
2. **Cancer**
   - **SEER (cases through 2006)**
   - **SEER-Medicare**
     (cases through 2005: request in progress)
3. **Medicare**
   - **Claims: Inpatient, Carrier (physician), Outpatient (clinic), etc.**
   - **Other: Hospital characteristics, Physician (limited)**
4. **Other: AMA (UPIN to NPI conversion), AHA (old)**

A Brief History of SAS®

- **1966-8: Anthony Barr develops SAS language**
- **1968: Barr and James Goodnight develop ANOVA and multiple regression procedures for SAS**
- **1973: John Sall joins the project**
- **1976: SAS Institute is incorporated by Barr, Goodnight and Sall**
- **1988: SAS v. 6 re-written in C for portability, adds support for UNIX, X11, SQL and RDBMS modern SAS era begins**
- **2008: SAS v. 9.2 (TS2M0)**
  by SAS Institute Inc., Cary, NC, USA
  (how to reference SAS in articles, grants, etc.)

SAS Book for Beginners

The Little SAS Book: A Primer 4th Ed. (2008) $49.95
by LD Delwiche and SJ Slaughter
SAS Press (available in MCW book store)

- **Getting Started Using SAS Software: 1.1-1.4**
- **Getting Your Data into SAS: 2.19-2.22**
- **Working with Your Data: 3**
- **Sorting, Printing and Summarizing Your Data: 4.1-4.7, 4.11, 8.1-8.4**
- **Modifying and Combining SAS Data Sets: 6.1-6.7, 6.9-6.12, 6.14**
- **ANOVA and Regression: 8.5-8.8**

Other SAS Resources

1. **More advanced SAS books**
   - **SAS Applications Programming: A Gentle Introduction (2008) by Frank Dilorio, Brooks/Cole Publishing**
   - **Applied Statistics and the SAS Programming Language 5th ed. (2005) by RP Cody and JK Smith, Prentice Hall**
   - **Statistical Analysis of Medical Data Using SAS (2005) by Geoff Der and Brian Everitt, CRC Press**
2. **SAS v. 8 manuals in PCOR (must not leave the office)**
3. **SAS v. 9.2 manuals online** `http://support.sas.com/documentation/cdl_main/index.html`
   plus "Knowledge Base/Tech Support"
4. **SAS-L mailing list** `http://www.listserv.uga.edu/archives/sas-l.html`

The SAS Language

- **Swiss Army Knife: data processing, statistical analysis, graphing/GIS, RDBMS access and more**
- **a combination of high-level, optimized PROCs and low-level DATAstep programming**
- **learn the SAS “way” of doing things**
- **use best-of-breed coding practices**
- **use short bits of PROC code whenever possible interlaced with DATAstep code for optimal results**

A SAS Program

- **a SAS program is a text file with a name ending in .sas: example1.sas**
- **to manually submit a SAS batch job from the UNIX command line**

```
godzilla % sas example1.sas &
```

- **generates a text log, .log, for notes and error messages and a text listing, .lst, for results: example1.log and example1.lst**
- **each SAS statement ends in a semicolon**
- **two styles of comments**

```
 * comment statement starts w/ asterisk ;
/* comment that is not a statement    */
```

Simple SAS Example: example1.sas

```
data surg;               * creating SURG dataset;
    input x1 x2 x3 x4 y; * reading in variables ;
    logx1 = log(x1);     * creating new variable;
    label x1 = 'Blood Clotting Score'
          x2 = 'Prognostic Index' /* descriptive */
          x3 = 'Enzyme Function Score' /* labels */
          x4 = 'Liver Function Score'
          y  = 'Survival Time';
  * x1  x2   x3   x4    y  1st x4 missing; lines;
   6.7  62   81   .    200
... 52 more lines of data not shown ...
   8.8  78   72  3.20  313
;    * do-nothing command denotes end-of-data;
run; * execute data/run block                ;
```

Log Notes: example1.log

```
1  The SAS System  15:32 Wednesday, May 13, 2009
...
32        * x1  x2   x3   x4    y  1st x4 missing;

NOTE: The data set WORK.SURG has 54 observations
and 6 variables.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.01 seconds

87        ;    * do-nothing denotes end-of-data;
88        run; * execute data/run block       ;
89
```

Error Message: example1.log

```
32           * x1  x2   x3   x4    y ;  fines;
                                        _____
                                        180
33            6.7  62   81   .    200

              ___
              180
ERROR 180-322: Statement is not valid or it is used
out of proper order.
... lines omitted ....
ERROR: No DATALINES or INFILE statement.
NOTE: The SAS System stopped processing this step
because of errors.
NOTE: SAS set option OBS=0 and will continue to
check statements.
```

Summary of Quantitative Variables: example1.sas

```
* procedures (or PROCs) operate on datasets    ;

proc contents data=surg; * describes dataset   ;
      * DATA= defaults to last dataset created ;
run;  * execute proc/run block                 ;

proc print; * creates listing for variables    ;
run;        * of last dataset created          ;

proc univariate;      * calculates stats       ;
    var logx1 x2-x4;  * variable list          ;
*   var _numeric_;    * all numeric variables  ;
run;
```

Summary of Quantitative Variables: example1.lst

```
The UNIVARIATE Procedure
    Variable:  logx1

 N                      54
 Mean             1.716764

 100% Max         2.415914
 95%              2.174752
 90%              2.041220
 75% Q3           1.871802
 50% Median       1.757858
 25% Q1           1.609438
 10%              1.308333
 5%               1.223775
 0% Min           0.955511
```

Summary of Qualitative Variables: example1.sas

```
proc format;   * creates value groupings for vars ;
    value cat
         0- <50='~<50' /* ~ is last ASCII char  */
        50-  75=' 50-'
        75-high='>75'; /* 75 is included in 50- */
run;

proc freq; * calculates freqs/pcts/stats          ;
    format x2 x3 cat.; * group variables w/ format;
    tables x2 x3;      * create freq/pct summaries;
run;
```

Summary of Qualitative Variables: example1.lst

```
                 The FREQ Procedure
                  Prognostic Index

                                    Cum.         Cum.
  x2   Freq      Percent            Freq        Percent
---------------------------------------------------------
˜<50      8        14.81               8          14.81
 50-     31        57.41              39          72.22
>75      15        27.78              54         100.00
```

Summary of Qualitative Variable Assocation: example1.sas

```
proc format;
    value surv
        0  −<100=′<100′
        100− 250=′ 100−′
        250−high=′>250′
    ;
run;

proc freq;
    format x2 x3 cat. y surv.;
    tables y⋆(x2 x3) / chisq;
⋆ Pearson Chi−squared Test of Independence;
run;
```

Summary of Qualitative Variable Assocation: example1.lst

```
                Table of y by x2

y(Survival Time)       x2(Prognostic Index)

Freq      |~<50     | 50-     |>75      |  Total
----------+---------+---------+---------+
<100      |       6 |       4 |       1 |     11
----------+---------+---------+---------+
 100-     |       2 |      23 |       6 |     31
----------+---------+---------+---------+
>250      |       0 |       4 |       8 |     12
----------+---------+---------+---------+
Total             8        31        15       54
```

Summary of Qualitative Variable Assocation: example1.lst

```
                The FREQ Procedure

         Statistics for Table of y by x2

Statistic        DF       Value      Prob
-----------------------------------------
Chi-Square        4     27.2514    <.0001

 WARNING: 56% of the cells have expected counts
 less than 5. Chi-Square may not be a valid test.
```

Multiple Linear Regression: example1.sas

```
proc glm;
    class x2 x3;
* fit as qualitative covariates                 ;
* last alphanumeric order group is the intercept;
* hence, a group starting with ~
             will always represent the intercept;
    format x2 x3 cat.;
    model y=logx1 x2-x4 / solution;
run;
```

Multiple Linear Regression: example1.lst

```
Dependent Variable: y  Survival
                Sum of       Mean
Source   DF    Squares      Square   F Value  Pr > F

Model     6   763559.297  127259.883 16.47   <.0001

Error    46   355364.024    7725.305

Corrected
Total    52  1118923.321
```

Multiple Linear Regression: example1.lst

```
Source DF  Type III SS  Mean Square F Value  Pr > F

logx1   1   31719.5895   31719.5895   4.11   0.0486
x2      2  124208.0011   62104.0005   8.04   0.0010
x3      2   96388.4192   48194.2096   6.24   0.0040
x4      1   52386.1755   52386.1755   6.78   0.0124
```

Multiple Linear Regression: example1.lst

```
                                Standard
Parameter          Estimate      Error        t     Pr > |t|

Intercept         -300.27 B    100.39     -2.99      0.0045
logx1              114.69       56.60      2.03      0.0486
x2         50-      44.88 B     36.16      1.24      0.2209
x2         >75     153.32 B     43.44      3.53      0.0010
x2         ˜<50      0.00 B       .          .         .
```