Old But Not Obsolete:
Undocumented SAS® Procedures

Barbara B. Okerson, Ph.D., CPHQ
Health Management Corporation (HMC)

Abstract

Proc SPELL? Proc NEIGHBOR? Proc BROWSE?

Through the years a number of SAS procedures have disappeared from the manuals, mainly because their functionality was picked up by newer, more robust, procedures and features of the SAS system. But sometimes, simpler can be better. This presentation looks at these and other procedures of the past that are not in the current documentation but still work and addresses situations where they can yet be useful.

HMC

- Experienced provider of total health solutions
- Care management
  - Chronic condition support
  - Lifestyle management
  - Complex condition care
  - Prevention
- Goal - improve the health and financial outcomes of clients

Life² the next level

360°Health A Revolution in Care.

FULL CIRCLE HEALTH UniCare

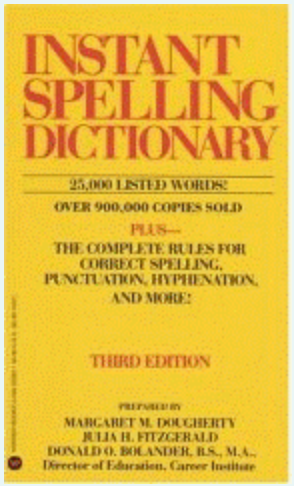
HMC building healthier lives

Procedures

- Useful Obsolete Procedures
  - Proc BROWSE
  - Proc SPELL
  - Proc NICKNAME
  - Proc NEIGHBOR
- Useful Obsolete Procedures (with cautions)
  - Proc EDITOR
  - Proc DELETE

Proc SPELL

- SAS Spell Checker
  - History
    - SAS V6
    - Lattice C Compiler
  - Default dictionary included
  - User created dictionary
    - Create a *text* file with the custom word list.
    - Put each word on separate line.
    - Point to SAS dictionary catalog (if updating) or create new catalog.
    - Point to location of custom word list.



HMC
building healthier lives

Proc SPELL Example

```
Proc SPELL words="r:\bokerson\sesug2007\condition.txt" create
    dict=work.mycatalog.spell; run;
Proc PRINTTO print="r:\bokerson\sesug2007\sp_cond.txt" new; run;
Proc SPELL wordlist="r:\bokerson\sesug 2007\spellcase.txt" verify
    dict=work.mycatalog.spell; run;
Proc PRINTTO print=print; run;
```

Custom Word List

AST
CAD
CHF
COP
DIA
MAT

Unrecognized word	Freq	Line(s)
DIAB	4	14, 32, 39, 234
CHHF	1	18
COPD	5	39, 234, 235, 236, 241
HF	1	349

HMC
building healthier lives

Proc BROWSE

- Read SAS data sets
  - Interactive (obsolete with explorer window)
  - Batch
- Read only access
- Statements
  - FIND – searches a range for occurrences of a variable.
  - LOCATE – searches a range for occurrences of a single value.
  - NAME – indicates a variable to search for matches.
  - SEARCH – searches for occurrences of a string or strings.
  - STRING – names variable or variables to be used by SEARCH.
  - LIST – lists the values of one or more variables in a range.

Proc BROWSE Example

- Batch program invoked in Windows Explorer:
  - Create and save Proc Browse SAS program.
  - Right-click on the SAS program (with .sas extension).
  - From pop-up menu, select: *Batch Submit With SAS 9.1.*
- Results file (.log) written to .sas program folder
- SAS Program

```
Libname r 'r:\bokerson';
Proc BROWSE data=r.medical;
NAME MemberID;
LOCATE 1,last 144490885;
Run;
```

HMC
building healthier lives

Proc BROWSE Results

```
      NOTE: Copyright (c) 2002-2003 by SAS Institute Inc., Cary, NC, USA.
NOTE: SAS (r) 9.1 (TS1M3)  Licensed to HEALTH MANAGEMENT CORP, Site XXXXXXXXXX.
NOTE: This session is executing on the NET_ASRV  platform.
NOTE: SAS 9.1.3 Service Pack 4
NOTE: SAS initialization used:
      real time           0.45 seconds
      cpu time            0.28 seconds

1          libname r 'r:\bokerson';
NOTE: Libref R was successfully assigned as follows:
      Engine:        V9
      Physical Name: r:\bokerson
2          proc browse data=r.medical;
NOTE: Welcome to the BROWSE Procedure. Begin entering commands.
3          name memberid;
4          locate 1,last 144490885;
NOTE: Found at OBS 1166639.
5          Run;
NOTE: PROCEDURE BROWSE used (Total process time):
      real time           10.31 seconds
      cpu time            3.37 seconds

NOTE: Exiting from the BROWSE Procedure.

NOTE: SAS Institute Inc., SAS Campus Drive, Cary, NC USA 27513-2414
NOTE: The SAS System used:
      real time           10.78 seconds
      cpu time            3.65 seconds
```

HMC
building healthier lives

Proc NICKNAME

- Purpose: allow SAS administrators to use SAS aliases for adding new modules.
- Usage: check available SAS engines and access methods.
- Proc NICKNAME Options:
  - ACCESS or AMETHOD – restricts search to access methods.
  - ENGINE – restricts alias list to engines.
  - CATALOG – restricts search to catalog specified. Default is sashelp.core.
  - FUNCTON - sets *ffalias* as default for function aliases.
  - FORMAT – sets *ffalias* as default for format aliases.
  - INFORMAT - sets *ffalias* as default for informat aliases.

Proc NICKNAME Example

SAS Code:

```
Proc NICKNAME eng;
Run;
```

Current Catalog: SASHELP.CORE

		Nickname	Module	Type	Fileformat	Description
P	M	ACCESS	SASIOMDB	ENG	7	SAS/ACCESS Interface to PC Files
	M	ACCESS99	SASECRSP	ENG		Read engine for CRSP ACCESS97 database
	M	BASE	SASE7	ENG	7	Base SAS I/O Engine
P	M	BLOOMBRG	SASIOBLB	ENG	9	SAS/Access Interface To Bloomberg
P	M	BMDP	SASBMDPE	ENG	607	BMDP Save file engine
P	M	CRSPACC	SASECRSP	ENG		Read engine for CRSP ACCESS97 database
P	M	CVP	SASECVP	ENG	9	Character Variable Padding Engine
P	M	DB2	SASIODBU	ENG	7	SAS/ACCESS Interface to DB2
P	M	EXCEL	SASIOXLS	ENG	7	SAS/ACCESS Interface to PC Files
P	M	FAMECHLI	SASEFAME	ENG		Seamless libname interface to FAME db
P	M	HAVERDLX	SASEHAVR	ENG	9	Read engine for Haver Analytics DLX db
P		IMDB	SASEIMDB	ENG	9	In Memory Database Engine
P	M	META	SASIOMET	ENG	7	Metadata engine
P	M	MYSQL	SASIOMYL	ENG	7	SAS/ACCESS Interface to MySQL
P	M	ODBC	SASIOODB	ENG	7	SAS/ACCESS Interface to ODBC
P		OLAP	SASEOLAP	ENG	9	SQL Passthru Engine for OLAP
P	M	OLEDB	SASIOOLE	ENG	7	SAS/ACCESS Interface to OLE DB
P	M	ORACLE	SASIOORA	ENG	7	SAS/ACCESS Interface to Oracle
P	M	OSIRIS	SASOSIRI	ENG	607	OSIRIS Data File engine
P		R3	SASIOSR3	ENG	9	SAS Engine for SAP R/3
P	M	REMOTE	SASIORMT	ENG	7	SAS/SHARE Remote access engine
P	M	REMOTE8	SASI8RMT	ENG	7	SAS/SHARE V8 Remote access engine
P	M	REUTERS	SASEREUT	ENG	612	Reuters financial market data interface
	M	SASIOOS2	SASIODBU	ENG	7	SAS/ACCESS Interface to DB2
P	M	SPDE	SASSPDE	ENG	7	Scalable Performance Data Engine
P	M	SPSS	SASSPSS	ENG	607	SPSS Save File engine
P		SQLVIEW	SASESQL	ENG	607	SQL view engine
	M	SXLE	SASEXML	ENG	8	W3C XML input/output engine
P	M	SYBASE	SASIOSYB	ENG	7	SAS/ACCESS Interface to Sybase
P	M	TERADATA	SASIOTRA	ENG	8	SAS/ACCESS Interface to Teradata
P		TRACE	SASETRC	ENG	7	Version 7 trace engine
P	M	V6	SASEB	ENG	607	Base SAS I/O Engine
P	M	V604	SASIO602	ENG	606	Base SAS I/O Engine - 6.06 defaults
	M	V607	SASEB	ENG	607	Base SAS I/O Engine
	M	V608	SASEB	ENG	607	Base SAS I/O Engine
	M	V609	SASEB	ENG	607	Base SAS I/O Engine
	M	V610	SASEB	ENG	607	Base SAS I/O Engine
	M	V611	SASEB	ENG	607	Base SAS I/O Engine
	M	V612	SASEB	ENG	607	Base SAS I/O Engine
	M	V7	SASE7	ENG	7	Base SAS I/O Engine
	M	V701	SASE7	ENG	7	Base SAS I/O Engine
	M	V8	SASE7	ENG	7	Base SAS I/O Engine
P	M	V9	SASE7	ENG	7	Base SAS I/O Engine
P	M	XML	SASEXML	ENG	8	W3C XML input/output engine
P	M	XPORT	SASV5XPT	ENG	607	Version 5 transport datasets

Proc NEIGHBOR

- Non-parametric nearest neighbor discriminant analysis
- Appropriate for radically non-normal distributions (disease incidence/prevalence)
- Valuable in avoiding assumptions
- Classification options:
  - Classified by proportion of k nearest neighbors
  - Classified by proportion of nearest neighbors times a prior class probability

Proc NEIGHBOR Options

Proc NEIGHBOR Statements:

- CLASS – defines classes (required).
- VAR – lists variables to be included.
- ID – identifies value used for observation ID in classification results.
- PRIORS – signifies prior probabilities unequal.
- TESTCLASS – names test data set variable used for misclassification.
- TESTID – names observation ID variable for classification results.
- BY – requests separate analyses for By groups.

Proc NEIGHBOR Statement Options:

- IDENTITY – specifies use of Euclidean distances.
- K – specifies k-value for k-nearest neighbor rule.
- LIST – prints classification results by observation.
- LISTERR – prints only misclassified observations.
- THRESHOLD – specifies minimum possible posterior probability for classification.
- TESTLIST – lists all observations in the *testdata=* data set.
- TESTLISTERR – lists only misclassified observations in *testdata=* data set.

Proc NEIGHBOR Example

SAS Code:

```
Proc NEIGHBOR k=1 data=ckd_ID
     testdata=testckd testlist;
Class CKD;
TestClass CKD;
TESTID memberid;
Var gender agegroup lab1-lab5;
Run;
```

Results:

The DISCRIM Procedure
Classification Results for Test Data: WORK.TESTCKD
Classification Results using Nearest Neighbor

Squared Distance Function

$$D^2(X,Y) = (X-Y)' \, COV^{-1} \, (X-Y)$$

Posterior Probability of Membership in Each ckd

$m_k(X)$ = Proportion of obs in group k in nearest neighbor of X

$$Pr(j|X) = m_j(X) \, PRIOR_j \, / \, SUM(m_k(X) \, PRIOR_k)$$

Posterior Probability of Membership in ckd

MemberID	From CKD	Classified into ckd		N	Y
2001	Y	N	*	1.0000	0.0000
2002	Y	N	*	1.0000	0.0000
2003	Y	N	*	1.0000	0.0000
2004	Y	N	*	1.0000	0.0000
2005	Y	Y		0.0000	1.0000

* Misclassified observation

Proc EDITOR

- Edit a SAS data set without a data step
- No new data set created- changes permanent
- Statements
  - REPLACE – changes values of specified variables in a specified range.
  - ADD – adds a new observation to the end of the data set.
  - DELETE – sets all variables in a specified range to missing.
  - DUP – duplicates observations for a specified range.
- Sample SAS code:

```
Libname r 'r:\bokerson\sesug 2007';
Proc EDITOR data=r.clientemployergroup;
Find 1, last employerdesc='National Bank';
Replace employerdesc='New Bank';
Run;
```

HMC
building healthier lives

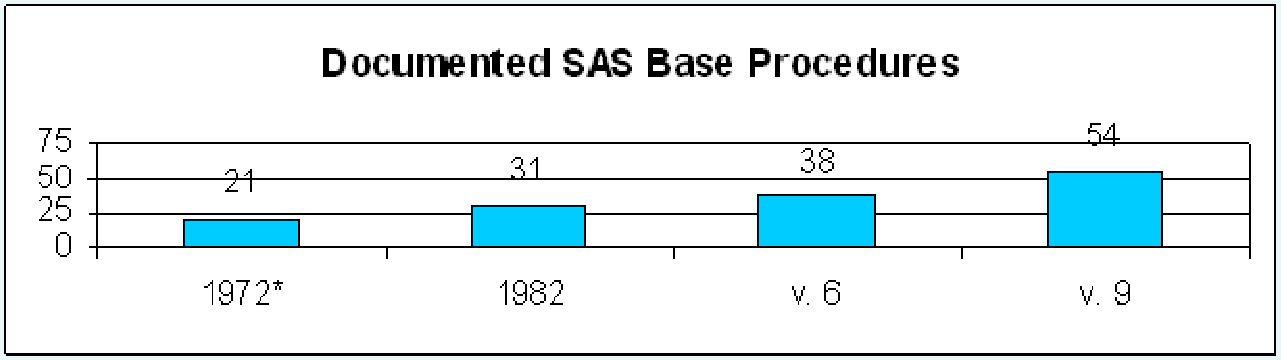
Proc DELETE

- Delete SAS data sets
- Can be more efficient than Proc DATASETS
- Caution: when used with no options entire work directory is permanently deleted
- Sample SAS code:

```
Proc DELETE data=work.test1 work.test2;
Run;
```

SAS Base Procedures

- Fully supported procedures continue to increase in numbers
- Base SAS increase from 21 to 54 procedures
- Even greater increase in supported options



* statistical procedures included in base.

Other Undocumented Gems

- Do over (arrays)
  - Implicit array – simpler code.
  - Documentation removed beginning with v7.1.
- Data set conversion with Proc DATASETS

```
libname try 'r:\bokerson\sesug 2007\newexcel.xls';
Proc DATASETS lib=work nolist;
  copy in=work out=try;
  select new;
  quit;
```

HMC
building healthier lives

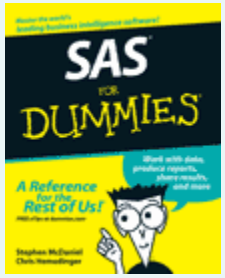
Conclusion

- Software is NOT like fine wines
- Generally not better with age
- Users demand more functionality and features
- Yet, sometimes the old is exactly what is needed

HMC
building healthier lives

References

- Jaffe, Jay. *Mastering the SAS System*. New York: Van Nostrand Reinhold, 1989.
- SAS Institute, Inc. *SAS Users' Guide: Basics, 1982 Edition*. SAS Institute Inc, Cary, NC, 1982.
- SAS-L archives. http://www.listserv.uga.edu/archives/sas-l.html.
- Service, Jolayne. *A User's Guide to the Statistical Analysis System*, North Carolina State University, 1972.



www.listserv.uga.edu

The University of Georgia

HMC
building healthier lives

Contact Info

Your comments and questions are valued and encouraged. For more information contact:

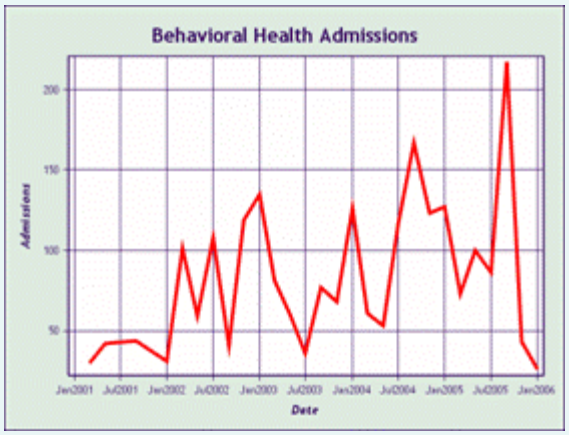
Barbara B. Okerson, Ph.D., CPHQ
Senior Health Information Consultant, Analytics
Health Management Corporation (HMC)
8831 Park Central Drive, Suite 100
Richmond, VA 23227
Office: 804-662-5287
Fax: 804-354-2468
Email: bokerson@choosehmc.com

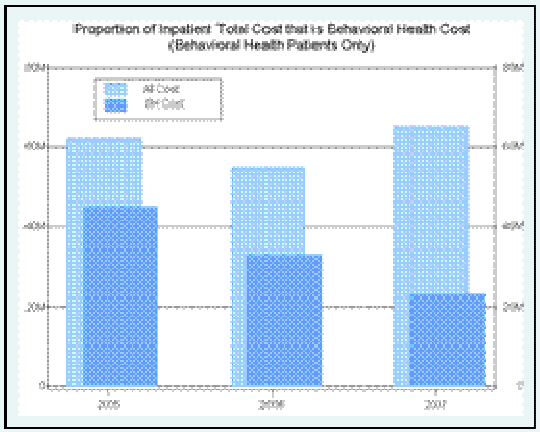
HMC
building healthier lives

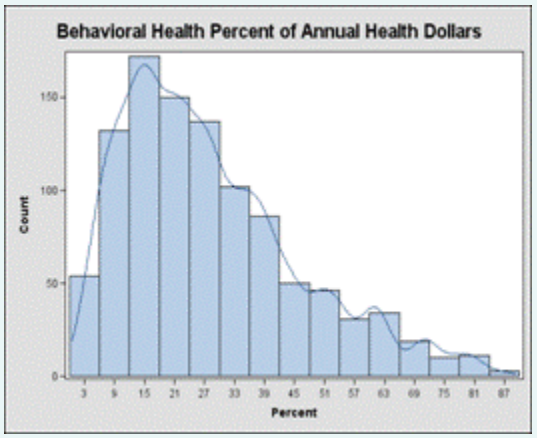
SAS GLOBAL FORUM 2008

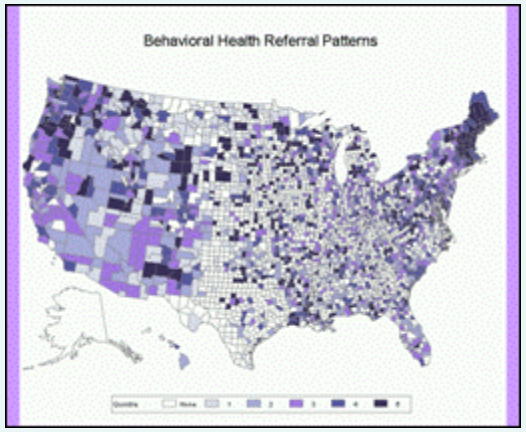
SAN ANTONIO TEXAS
MARCH 16-19 2008
Warren E. Stinson, Conference Chair

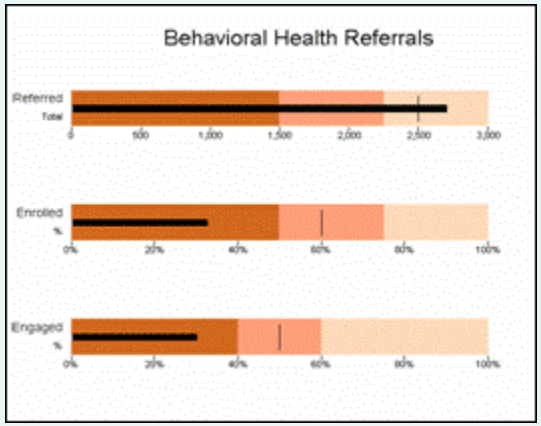
Using SAS® Graphics to Explore Behavioral Health Cost Risk











HMC
building healthier lives