



In this issue

- [Executive Update](#)
- Tips & Tricks:
 - [Using an Autocall Library](#)
 - [Exploring Data with Binary Variables and PROC FREQ/LIST](#)
 - [Shortening Your DO Step IN: Arrays](#)
 - [Running Simulations without Output](#)

Who We Are

The Vancouver SAS Users Group (VanSUG) is an informal group of SAS users in the Vancouver area who meet and share their knowledge. There are no fees – we simply meet every spring and fall to have presentations, networking, and fun! Check us out at vansug.ca.

Newsletter Inputs

This newsletter is for and by the local Vancouver SAS community. If you'd like to contribute to a future newsletter, please email us at vansug@gmail.com!

Executive Team

President:

Colleen McGahan
(BC Cancer Agency)

Vice President:

Suzanne Humphreys
(Rick Hansen Institute)

Other Members:

Scott Albrechtsen (BC Hydro)
 Núria Chapinal (BC Centre for Disease Control)
 Mei Chong (BC Centre for Disease Control)
 Nate Derby (Stakana Analytics)
 Jing Dong (BC Hydro)
 Julia Zhu (BC Centre for Excellence in HIV/AIDS)

Executive Update

We're very excited to offer something new: We now have a VanSUG blog (blog.vansug.ca) to foster discussion and problem solving between all of you regarding the BA problem put forward in today's meeting. We hope that many of you will post comments and solutions on this blog, so please take a look at it! Additionally, if you have any other ideas of how this site could be utilized to help networking and communication between VanSUG users, please don't hesitate to let one of the VanSUG Executive members know – or just blog your suggestion!

TIPS & TRICKS: Using an Autocall Library

Nadia Lesnikova, BC Centre for Disease Control

The autocall macro facility allows invoking a macro that has not yet been defined in a current SAS session. Any SAS macro stored in the autocall library can be used in a SAS program as needed. It can also be used by multiple users at the same time.

Follow these two simple steps to use autocall macro facility:

1. *Set up SAS autocall library.* One or more folders can be designated as a SAS autocall library. Each folder must contain a macro definition, saved in a file with the same name as the name of the defined macro. Each folder may contain as many library members as you need.

2. *Specify the following system options:*

MAUTOSOURCE: Turns on autocall macro facility

SASAUTOS= ("library-1"<, . . . , "library-n">): Specifies the location of the autocall libraries. These are the locations (path names) where the macros are stored.

Example:

```
FILENAME m1b1 "O:\ABC\DEF\macrolib1";
FILENAME m1b2 "O:\ABC\DEF\macrolib2";
OPTIONS MAUTOSOURCE SASAUTOS=(m1b1, m1b2);
```

Once you run the lines of code above, all macros stored in the specified autocall libraries will be available to you at any time during the current SAS session. Use the macro name to call the macro from autocall library in any of your SAS programs:

```
%macro_test (param1, param2, param3)
```

Things to remember:

- Store only autocall library files in designated folders.
- Keep only one macro in any autocall library member.

The autocall library is a powerful facility. It helps to ensure consistency across all programs and faster updates. It's easy to use and maintain.

TIPS & TRICKS: Exploring Data with Binary Variables and PROC FREQ/LIST.

Dmitry Shopin: dshopin@cfenet.ubc.ca

Data exploration is an important step in data analysis that helps to make sense of your data, to decide how to proceed with data cleaning and what statistical methods and models to choose. One of the most usual things that data analysts are interested in is how variables are populated and if there is any pattern in the distribution of missing values between different variables. For example, when cleaning addresses, we'd like to know how many patients have neither postal code nor city in order to decide if we should ignore this missing information or have to try to impute it from some other sources.

We can do this very quickly using binary variables and PROC FREQ with the LIST option:

```
DATA address;
  SET address;
  is_postal = NOT MISSING(postal_code);
  is_city = NOT MISSING(city);
RUN;

PROC FREQ DATA=address;
  TABLES is_postal*is_city / LIST;
RUN;
```

TIPS & TRICKS: Shortening Your DO Step IN: Arrays

Gillian Frosst: gillian.frosst@phsa.ca

Arrays allow you to perform repetitive analysis across multiple variables (goo.gl/Bc7Pzj). One particular utility of arrays is reading across character variables to identify those beginning with a certain value or series of values. In the DO group processing of an array program, the SUBSTR function is adequate when your value(s) of interest are the same length (e.g., all values beginning with "A", "B" or "C"). However, coding quickly becomes tedious when you wish to identify records beginning with values of different lengths (e.g., all codes beginning with "A", "B1" or "C12"). The IN: operator simplifies coding by searching for character values of different lengths in a single step.

Suppose you have a dataset containing serial numbers for 500,000 records, where each record can have up to 25 serial numbers associated with it. Serial numbers are provided as variables serial_num1 to serial_num25 in your dataset. You would like to flag all records that have at least one serial number beginning with "12," "345," or "6789."

Using the SUBSTR function, your code would look something like this:

```
ARRAY serial_num[25] serial_num1-serial_num25;
DO a=1 to 25;
  if SUBSTR(serial_num[a],1,2) = "12" then flag=1;
  if SUBSTR(serial_num[a],1,3) = "345" then flag=1;
  if SUBSTR(serial_num[a],1,4) = "6789" then flag=1;
END;
```

However, using the IN: operator, your code would look something like this:

```
ARRAY serial_num[25] serial_num1-serial_num25;
DO a=1 to 25;
  if serial_num[a] IN: ("12","345","6789") then flag=1;
END;
```

TIPS & TRICKS: Running Simulations without Output

Mei Chong: mei.chong@telus.net

If you're running a simulation study and don't want to see the output from thousands of iterations, you can turn off all output from procedures by using the ODS SELECT command:

```
ODS SELECT NONE;

PROC procedure1;
PROC procedure2;
:
RUN;

ODS SELECT ALL;
```

Our next meeting will be in **Fall, 2014**. Check out the VanSUG website at vansug.ca later for more information!

Archives presentations and newsletters, as well as a link to scheduled SAS training courses held in Vancouver, can all be found on the VanSUG website at vansug.ca!

The **2014 Pharmaceutical Industry SAS Users Group (PharmaSUG)** conference will be held on June 1-4 in San Diego, CA. Information is at www.pharmasug.org.

The **2014 Western Users of SAS Software (WUSS) Conference** will be held in San Jose, CA on September 3-5. Come on down! Information can be found at wuss.org.

The **2014 MidWest SAS Users Group (MWSUG) Conference** will be held in Chicago, IL on October 5-7. Check it out at mwsug.org.

The **2014 Analytics Conference** will be held in Las Vegas, NV on October 20-21. Check it out at goo.gl/Zvidw2.