



In this issue

- [Executive Update](#)
- [Tips & Tricks:](#)
 - [MODE=include option](#)
 - [Summation by groups using Do Until](#)
 - [MODIFY statements](#)
 - [DLCREATEDIR option](#)

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:

Núria Chapinal
(BC Centre for Disease Control)

Mei Chong
(BC Centre for Disease Control)

Jing Dong
(BC Hydro)

Nate Derby
(Stakana Analytics)

Jeremy Hamm
(BC Cancer Agency)

Executive Update

Welcome to another Vancouver SAS Users Group meeting! After celebrating our ten-year anniversary this spring, we are eager to continue serving you, the SAS users of Vancouver, for another ten years. As always, please let us know at vansug@gmail.com, if you are interested in giving a presentation or writing a newsletter article. We would be happy to hear from you.

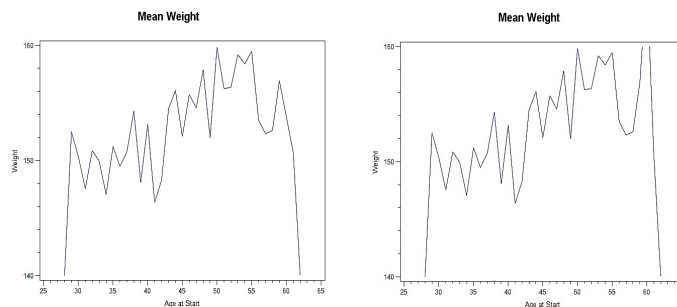
TIPS & TRICKS: See every data point with the MODE=include option

Nate Derby: nderby@stakana.com

When graphing with INTERPOL=join to join the data points, make sure you also include the MODE=include option to see all the data points when you restrict the range of the axis. Suppose you are looking at the mean weight from the sashelp.heart data set with the graph restricted to weights between 140 and 160 pounds.

```
SYMBOL1 INTERPOL = join;
  AXIS2 ORDER = ( 140 TO 160 BY 10 );

PROC GPLOT DATA = maxWeight;
  PLOT weight*age = 1 / VAXIS = AXIS2;
RUN;
```



You get the result above on the left. You are actually missing an important data point but can not see it. This is clear when you add MODE=include to your SYMBOL1 statement (see the result above on the right).

```
SYMBOL1 INTERPOL = join MODE = include;
```

This option includes all data points, even when they are outside the range of the graph. That is, a line is drawn to every data point (because of INTERPOL=join), even when that data point is outside of the graph. We get the result on the right, where we see that for age 33, the mean weight is above the maximal shown weight of 160 pounds. On the left graph, that data point was not shown at all, and there was a line drawn from the weight at age 32 to the one at age 33. We would have missed something very important.

TIPS & TRICKS: Summation by groups using Do Until

Yaling Yin: Yaling.Yin@bccancer.bc.ca

When doing cumulative summation by groups, it is common to have a SET statement preceding a conditional operation with FIRST or LAST variables generated by SAS automatically for group variables. Using DO UNTIL loop followed by a SET statement would also give summations across the group variables and return the last row per group with a group summation. Using the 'OF' in the brackets of the SUM function will ignore any missing values.

```
DATA newdata;
  total_cost = 0;
  DO UNTIL (last.patient_id);
    SET rawdata;
    BY patient_id;
    total_cost = SUM(OF total_cost per_claim_cost);
  END;
RUN;
```

TIPS & TRICKS: Use the MODIFY statement to update a data set

Min Yu: min.yu@bchydro.com

When we submit a DATA step with MODIFY statement, we are not only able to update all observations in a data set, but also can update only selected observations. MODIFY statement with BY statement as below will update a master SAS data set with values in a transaction data set by matching observations.

```
DATA master-data-set;
  MODIFY master-data-set transaction-data-set
  UPDATEMODE = MISSINGCHECK|NOMISSINGCHECK;
  BY key-variable;
RUN;
```

Where

- master-data-set is the name of the SAS data set that we want to modify;
- transaction-data-set is the name of the SAS data set with updated values;
- key-variable is the name of the variable whose values are matched in the master and transaction data sets;
- how missing values in the transaction data set are handled can be specified by using UPDATEMODE=option in the MODIFY statement.

When we use the MODIFY and BY statements to update a data set, WHERE processing starts at the top of the master set, finds the first match and updates it. If duplicate values of the BY variable exist in the master data set, only the first observation in the group duplicate values is updated. If duplicate values of the BY variable exist in the transaction data set, the transactions overwrite each other so that only the last transaction in the group is the result in the master data set.

TIPS & TRICKS: DLCREATEDIR Option

Jerome Buenaventura:

Jerome_img@yahoo.com

DLCREATEDIR option specifies to create a directory for a SAS library that is named in a LIBNAME statement if the directory does not exist.

If the path specified in the LIBNAME statement contains multiple components, SAS creates only the final component in the path. If any intermediate components of the path do not exist, SAS does not assign the specified path. For example,

```
OPTIONS DLCREATEDIR;
LIBNAME test "c:\temp\test";
```

when the above code executes, and c:\temp exists, SAS creates the test directory. If c:\temp does not exist, SAS does not create the test directory.

Our next user group meeting will be in **May 10, 2017**. Check out the [VanSUG website at vansug.ca](http://vansug.ca) later for more information!

The archived 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 **SAS Global Forum 2017** will be held in Orlando, FL on April 2-5, 2017. Information can be found at sasglobalforum.com.

The **Western Users of SAS Software (WUSS) Conference 2016** will be held in Long Beach, California on September 20-22, 2017. Information can be found at wuss.org.