

Interactive Programming Using Task in SAS® Studio

Suwen Li, Hoffmann-La Roche Ltd., Mississauga, ON

ABSTRACT

SAS® Studio is a web browser-based application with visual point-and-click user interface tasks that generate SAS code. Users can create their own tasks using the common task model and the Velocity Template Language. This paper illustrates how to conduct interactive programming by creating customized tasks in SAS® Studio.

INTRODUCTION

SAS® Studio is a SAS development application that user can access through web browser. It has predefined tasks that are XML and Apache Velocity code files to generate SAS code and format results. You can select a dataset from a drop down list, filter data, explore data, and build model in task. SAS code is generated in the code window while you are making the selections.

You can create your own tasks as well by editing the XML code to set up options that capture the inputs you want. Those inputs then pass into your code for execution. You can create different code for different analyses rather than creating complex macro program.

This paper will show how to develop SAS program interactively with customized tasks for survival analyses, swimmer plots, and ADSL data creation as examples.

SURVIVAL ANALYSIS

A customized task was created in Figure 1 to compute Kaplan-Meier curve using PROC LIFETEST. The task also has an option to estimate Hazard ratio using the Cox model.

When clicking the **Value** box, a list of distinct values of the variable selected in the **Variable 1** box will appears. By selecting different value from the **Value** drop-down list, you can perform analysis for different end points for example overall survival, progression free survival, etc.

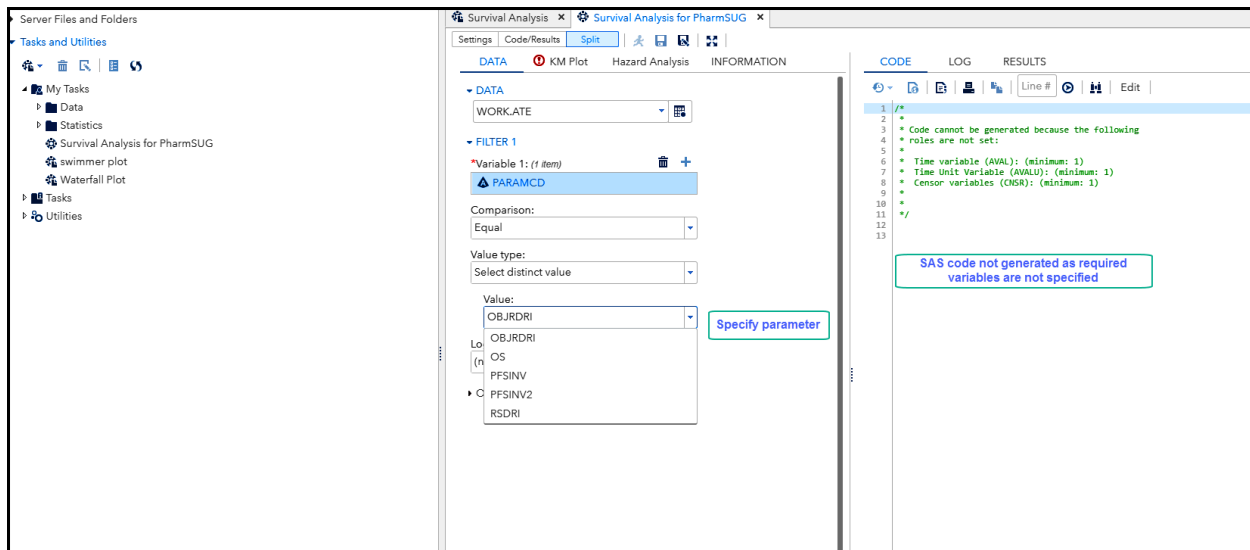


Figure 1 Select Endpoint for Analysis

When all the required variables for time and censor are entered, code is generated in Figure 2. The KM plot is displayed in Figure 3. As no strata variable is selected, only one survival function is displayed for all patients.

The screenshot shows the SAS Studio interface for a survival analysis. On the left, the 'ROLES' pane lists variables: AVAL (Time variable), AVALU (Time Unit Variable), CNSR (Censor variables), and Column (Strata variables). A red box highlights the 'Column' variable with the text 'Strata variable not specified'. The right pane shows the SAS code generated, including a PROC LIFETEST statement. A red box highlights the code with the text 'No STRATA statement in code'.

Figure 2 Select Variables

The screenshot shows the SAS Studio interface displaying the results of a Kaplan-Meier plot. The left pane shows the same variable selection as Figure 2. The right pane shows the 'RESULTS' tab with a Kaplan-Meier plot titled 'Kaplan-Meier Plot of Earliest Contributing Event to Investigator PFS With Number of Subjects at Risk'. The plot shows survival probability over time (weeks) with a table of subjects at risk and a summary table below.

Weeks	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60
AtRisk	17	17	17	14	14	9	8	3	3	2	2	1	1	1	1	0

	Median	95% CL
Survival	26	16.14 42.86

Figure 3 Kaplan-Meier Plot for All Patients

You can run the same analysis by group if you enter variable in the **Strata variable** box. For example, Figure 4 is Kaplan-Meier plot by treatment arm when the variable ARMCD is entered.

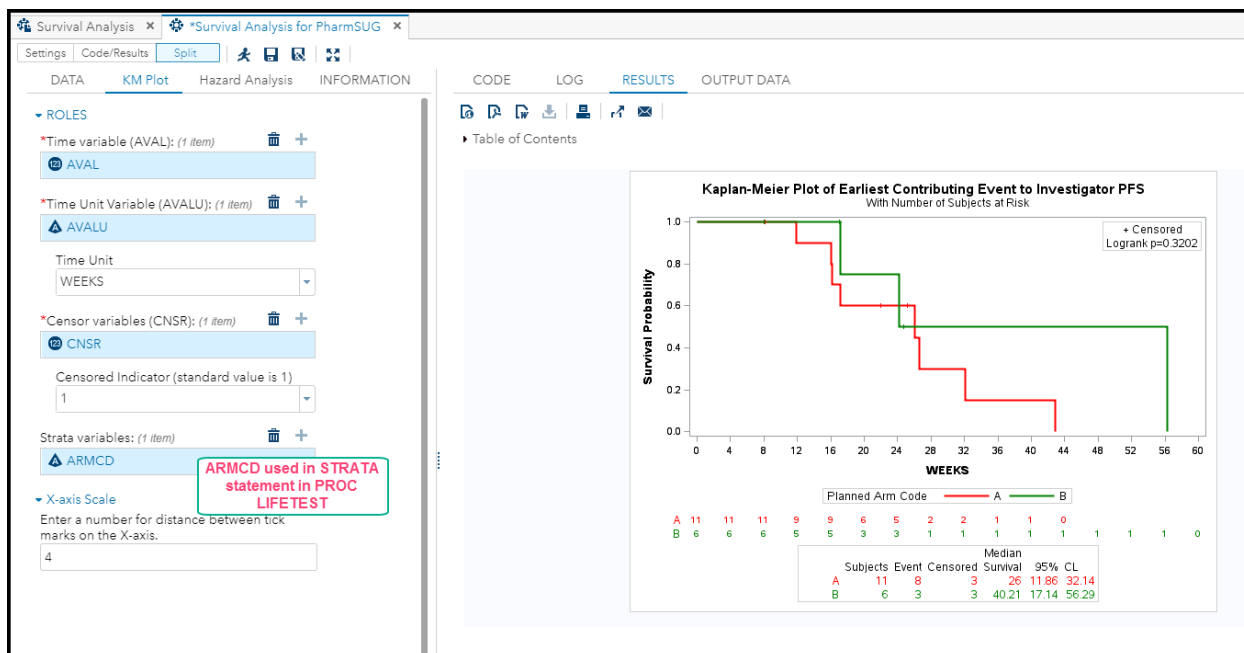


Figure 4 Kaplan-Meier Plot by Treatment Arm

If the **Hazard Analysis** box is checked, PROC PHREG to fit Cox regression is added into the code. Hazard ratio then is added in the summary in the Kaplan-Meier plot, see Figure 5 and Figure 6.

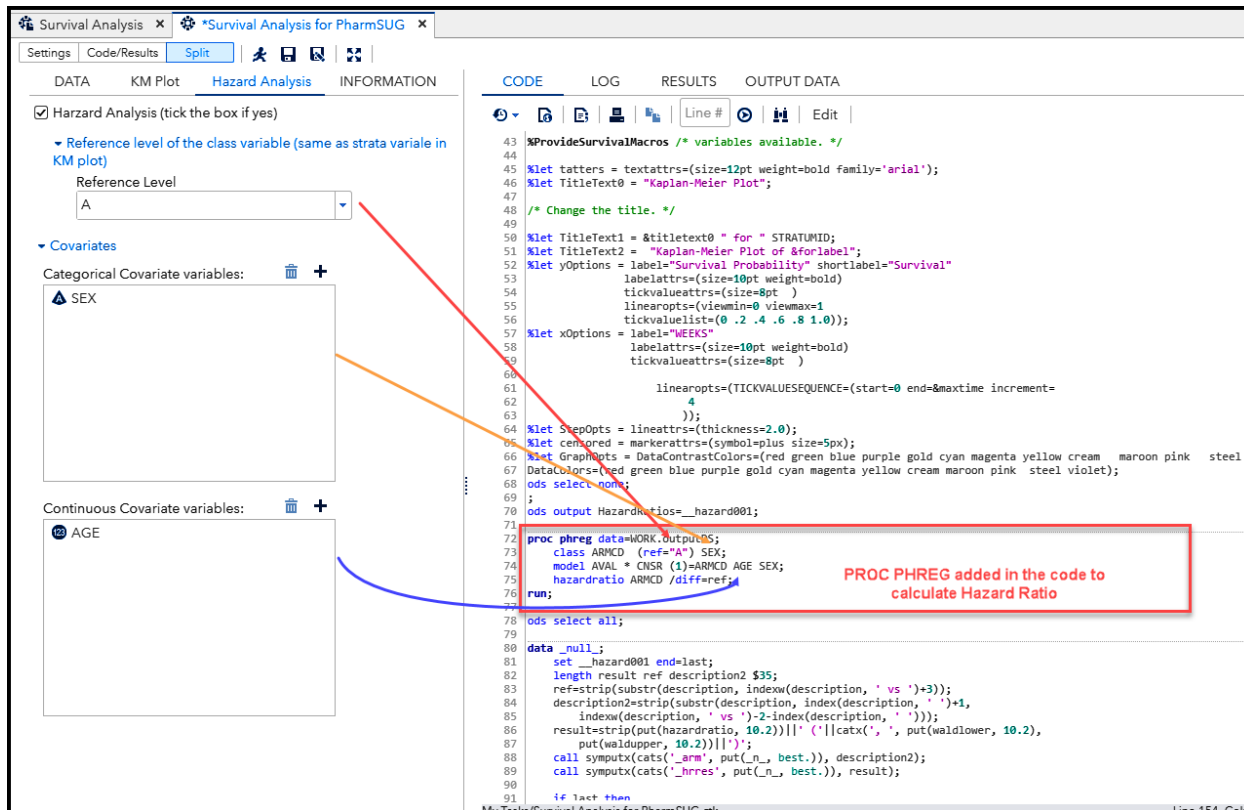


Figure 5 Estimating Hazard Ratio

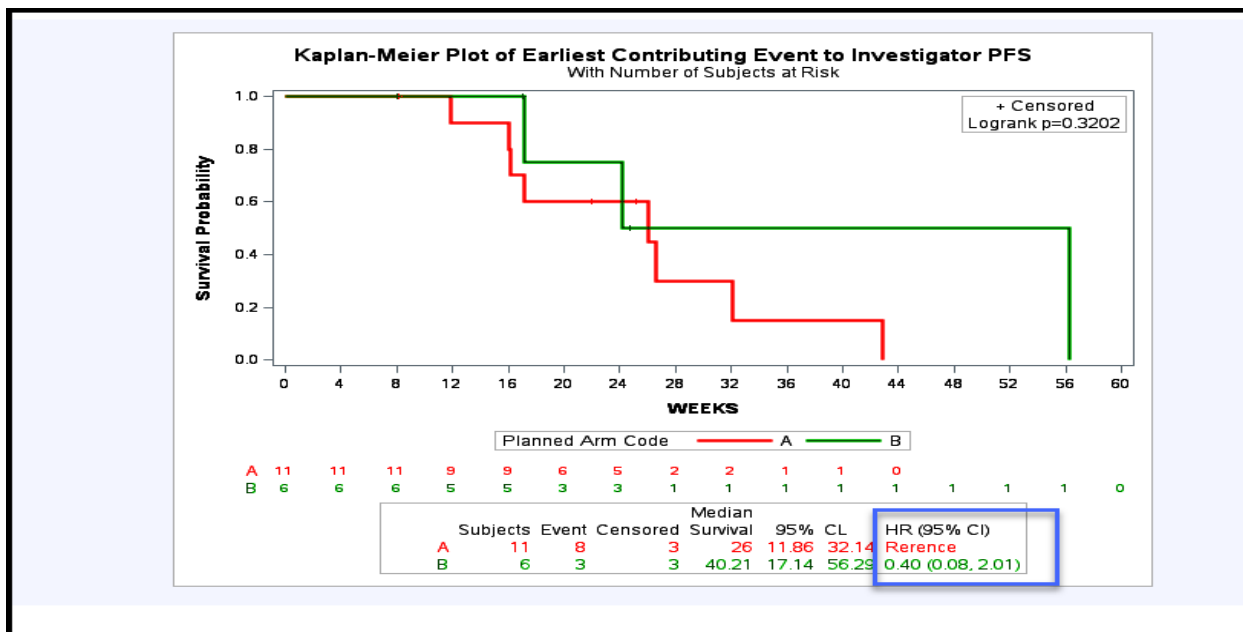


Figure 6 Kaplan-Meier Plot with Hazard Ratio in Summary

SWIMMER PLOT

The task in Figure 7 creates swimmer plot with custom data tips. The data tips can help QC and provide additional information which is not plotted in the figure for data review. The variables to be displayed in data points are specified in the box under **Display Data Tips at the Point of the Cursor** heading. Thus, the variable data that appears in the data tips can change dynamically on different output requests. The plot can be separated by different group if variable is selected in the **Variable** box.

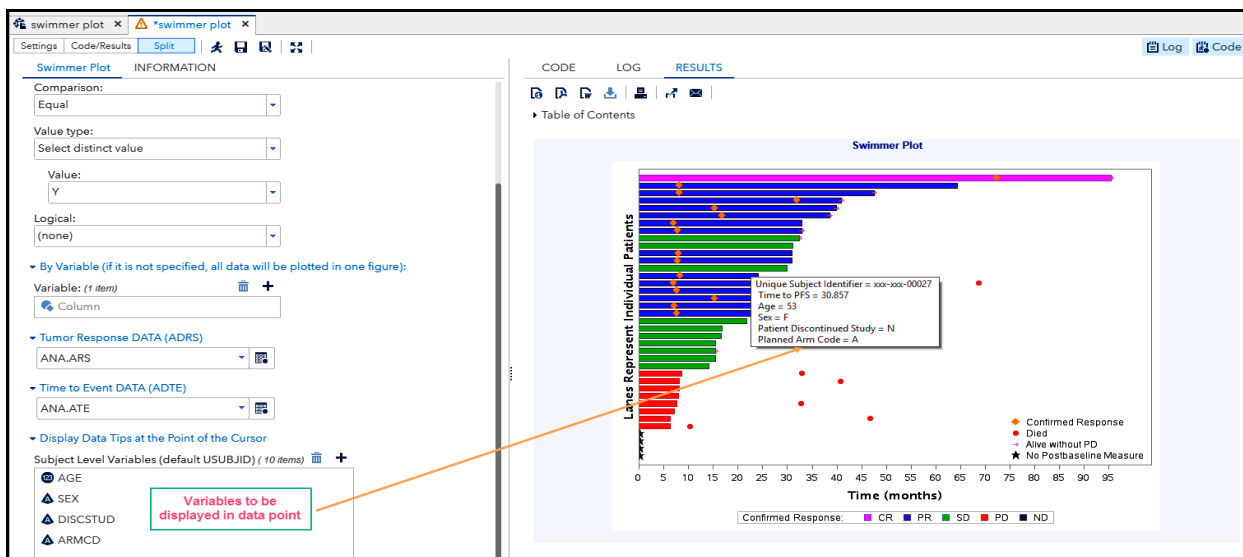


Figure 7 Swimmer Plot with Data Tips

ADSL

Some companies have standard data specifications with options for different scenario. For example, in below ADSL standard data specification template in Figure 8, there are 5 options for AAGE and 2 options for ITTFL.

	A	B	D	E	K
	Dataset	Description	Variable	Label	Derivation / Comment
1	ADSL	Subject Level Analysis Dataset	AAGE	Analysis Age	Option 1 (Default): Patient's age (years) at randomization. Set to integer part of (Randomization Date [ADSL.RANDDT] - Birth date [ADSL.BRTHDT] +1)/365.25).
33	ADSL	Subject Level Analysis Dataset	AAGE	Analysis Age	Option 2: Patient's age (years) at the initiation of study treatment. Set to integer part of (First Treatment Date [ADSL.TRSDT] - Birth Date [ADSL.BRTHDT] +1)/365.25).
34	ADSL	Subject Level Analysis Dataset	AAGE	Analysis Age	Option 3: Patient's age (mo) at randomization. Set to integer part of (Randomization Date [ADSL.RANDDT] - Birth date [ADSL.BRTHDT]+1)/30.4375).
35	ADSL	Subject Level Analysis Dataset	AAGEU	Analysis Age Unit	Set to "YEARS" if Analysis Age [ADSL.AAGE] is computed in years. Set to "MONTHS" if Analysis Age [ADSL.AAGE] is computed in months.
36	ADSL	Subject Level Analysis Dataset	ITTFL	Intent-To-Treat Population Flag	Option 1 (Default): Set to "Y" if patients Randomization Date: Start Date/Time of Disposition Event [DS.DSSTDTC] not equal to missing for a randomized study where Standardized Disposition Term [DS.DSDECOD] = "RANDOMIZATION".
43	ADSL	Subject Level Analysis Dataset	ITTFL	Intent-To-Treat Population Flag	Option 2: Set to "Y" if patients Enrollment Date: Start Date/Time of Disposition Event [DS.DSSTDTC] not equal to missing for a non-randomized study where Standardized Disposition Term [DS.DSDECOD] = "ENROLLMENT". Else set to "N"
44	ADSL	Subject Level Analysis Dataset	ITTFL	Intent-To-Treat Population Flag	

Figure 8 ADSL Data Specification

Below is the standard SAS program including code for different options but commenting out non-default ones. Programmer then needs to modify the code to select specific options based on his/her study. It is also not easy to validate the program with so many codes commented out.

```

***** ITTFL for Randomized Study *****;
*****Option 1 (default) *****;
data ittf1 (keep = usubjid ittf1);
  attrib ittf1 label="Intent-To-Treat Population Flag" length=$1;
  set sdtm.ds (where=(dsdecod="RANDOMIZATION" and ^missing(dsstdtc)));
  ittf1 = 'Y';
run;
data qc_adsl;
  merge qc_adsl(in=inadsl)
        ittf1 ;
  by usubjid;
  if inadsl;
  if missing(ittf1) then ittf1 = "N";
run;
***** ITTFL for Non-Randomized Study *****;
*****Option 2 *****;
/*data ittf1;
  attrib ITTFL label="Intent-To-Treat Population Flag" length=$1;
  set sdtm.ds (where=(DSDECOD="ENROLLMENT" and ^missing(DSSTDTC)));
  ITTFL = 'Y';
run;*/
*****Option 1 (default) *****;
***** age_analysis_refdt = randdt *****;
***** age_analysis_unit = Y *****;
aage = int((randdt - brthdt + 1)/365.25);
aageu = "YEARS";
***** Option 2 *****;
***** age_analysis_refdt = trtsdt *****;
***** age_analysis_unit = Y *****;
*aage = int((trtsdt - brthdt + 1)/365.25);
*aageu = "YEARS";
***** age_analysis_unit = M *****;
*aage = int((trtsdt - brthdt + 1)/30.4375);
*aageu = "MONTHS";

```

Although a SAS macro can be developed to accommodate all options, the macro program is not easy to read. Especially when we submit the program used to create ADaM dataset, we hope the program is simple and readable.

With SAS® Studio customized task, programmer can create readable code for his/her study without spending much time to modify the standard ADSL program. He/she only needs to point and click some options in the interface task.

In Figure 9, the SAS program has a block of code to derive ITTFL and AAGE using randomization date RANDDT only for default. There is no code for other options in it; in Figure 10, when options 2 are selected for ITTFL and AAGE, the code automatically switches to use enrollment date to derive ITTFL and use TRTSDT for AAGE calculation. Changing program header and libname is handled in this task as well. It will ultimately reduce program development time. On top of that, it is very easy to validate the program with different option combination.

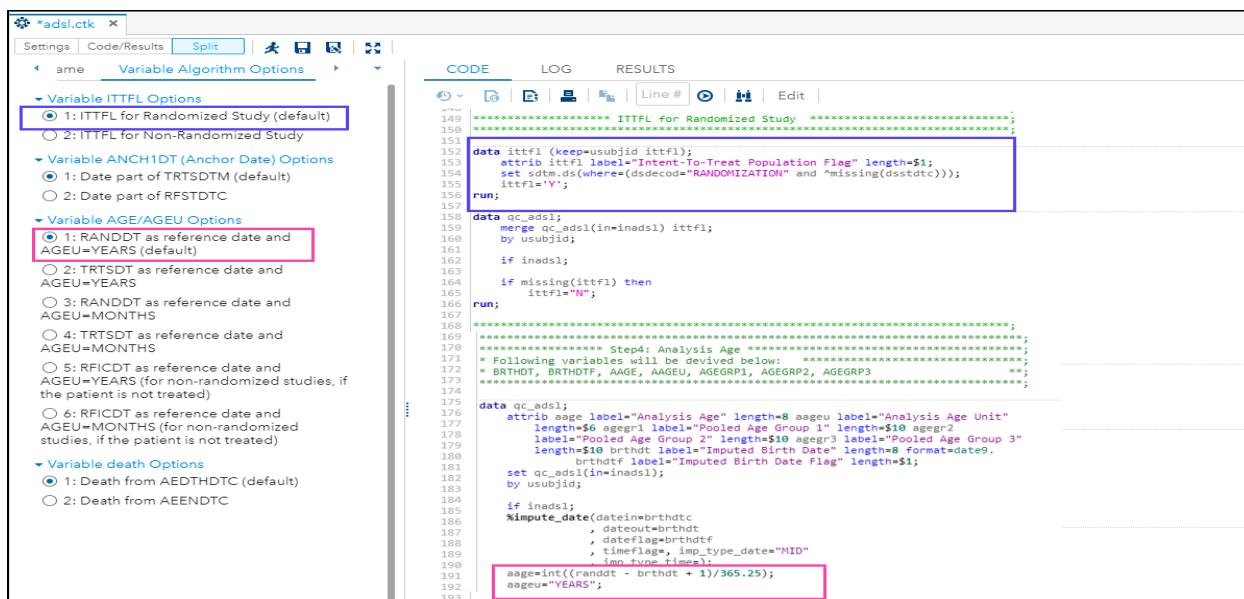


Figure 9 ADSL Program with Default Options

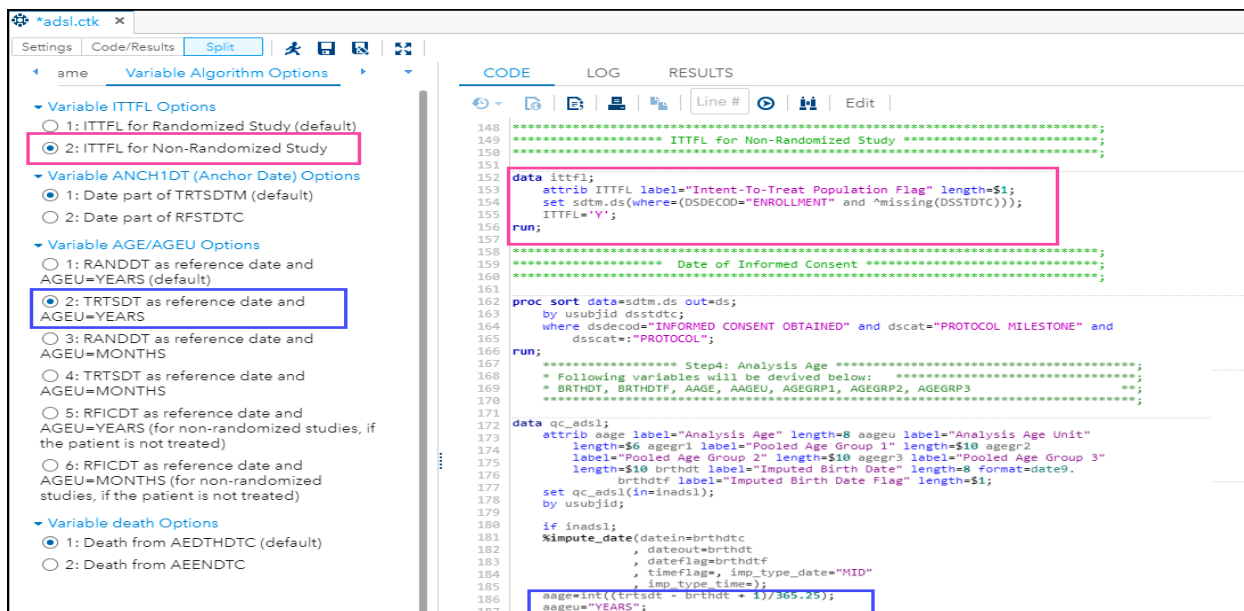


Figure 10 ADSL Program with Options 2 for ITTFL and AAGE

CONCLUSION

With custom task in SAS® Studio, you can create SAS program interactively for different requests or options by point-and-click only. It saves your time to modify code or develop complicated SAS macro.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Suwen Li
Hoffmann-La Roche Limited
7070 Mississauga Road
Mississauga, ON L5N 5M8, Canada

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.