**FDA JUMPSTART SCRIPT Liver\_v2.sas**

**TESTING USING FDA DATA AND ALSO ACCENTURE DATA**

**PROGRAM NAME**: Liver\_v2.sas (Liver Labs General)

.

**TESTING AUTHORS**: Eric Qi, Merck.

**SAS VERSION**: 9.3

**DESCRIPTION**: The program was tested in PC SAS with a Merck Laptop using FDA-provided data and also One Merck study data.

**TESTING STEPS**

1. All the necessary files were stored into C:\phuse\LiverLabsGeneral. Three test cases were created, case1 using Merck data; case2 using FDA-provided data; case3 using modified FDA-provided data Changed some DILI lab values.
2. CHANGES MADE TO THE SCRIPT TO MAKE IT RUN:

Added: %let root=C:\phuse\LiverLabsGeneral;

Updated line 114: %let studypath = &root.\case3\indata;

Updated line 122: %let saspath = &root.\SASIncludePrograms;

Updated line 123: %let utilpath =&root.\ZZ\_Utilities;

Updated line 125: %let templatepath = &root.\Templates;

Updated line 128: %let outpath = &root.\case3\output;

1. Run the program.
2. Check that LOG file is free of ERRORS, WARNINGS, and other suspicious messages.
3. Check that spreadsheet **Liver\_Labs.xls**

**TESTING CASES**

|  |  |  |
| --- | --- | --- |
| Case description | Pass/failure | note |
| Using Merck data | pass | Create correct result. Compared with outputs created by Merck macros. However, got the following error message:  NOTE: Invalid second argument to function SUBSTR at line 17 column 97.  a=1 b=1 lbtest=ALP LBSTAT=Missing LBREASND=M count=34 FIRST.lbtest=1 LAST.lbtest=1 stat\_reasn=Missing/M order=1 \_ERROR\_=1  \_N\_=1  It is hard to debug. Can’t find reason. The message from sub-maacro liver\_lbstresn\_missing. In this macro used a macro variable &lb\_lbreasnd, can’t find out where this macro variable created. |
| Using FAD data | pass | A warning in log when there are no error  NOTE: The data set WORK.LB\_ERR\_SV\_ABNORMAL has 0 observations and 66 variables.  WARNING: The variable col1 in the DROP, KEEP, or RENAME list has never been referenced. |
| Using FDA modified data, Changed some DILI lab values | pass | Same as case 2 above. |

**SUGGESTIONS ON POSSIBLE IMPROVEMENTS**

* Do not delete all global macro variables. User needs using global macro variable, for example to define root path.

data macrovar;

set sashelp.vmacro(keep=scope name where=(scope='GLOBAL' & name not in ('RUN\_LOCATION', 'ROOT'))); run;

data \_null\_; set macrovar; call execute('%symdel '||trim(left(name))||';');

run;

* If intend to share these scripts with industry, do not make “the installation of SAS PC Files Server” as required. Make it optional and easy to change. The change should be able to make in %params. Currently we have to go through many program files.
* Change line 155: proc printto print="&saspath.\liveroutput.lst"; run; quit; into:

proc printto print="&outpath.\liveroutput.lst" log=”&outpath.\liveroutput.log” new; run; quit;

because saspath is the SASIncludePrograms folder.

* Suggest to make file name same as macro name. One macro per file. This will make user easy to find file/macro.
* Suggest to use SAS SASAUTOS options to point to the macro locations, rather than using %include. Example code:

options mrecall validvarname=upcase mautosource mautolocdisplay;

filename fcurmdir " &saspath.";

filename fplyms " & utilpath.";

options sasautos=(

fcurmdir fplyms sasautos

);

* It would be useful to provide a listing of subjects that meet DILI criteria.
* It would be useful if we can use Charts to locate subject ID with extreme lab values.