A Macro that can Search and Replace String in your SAS Programs

Ting Sa, Cincinnati Children's Hospital Medical Center, Cincinnati, OH

ABSTRACT

In this paper, a SAS macro is introduced that can search and replace any string in the SAS programs. To use the macro, the user only needs to pass the folder name, the search string to it. If the user wants to use the replacement function, the user also needs to pass the replacement string to the macro. The macro will check all the SAS programs in this folder and the subfolders to find out which files contain the search string. The macro will generate new SAS files for the replacement so that the old files will not be affected. An html report will be generated by the macro to include the original file locations, the line number of the SAS codes that contain the search string and the SAS codes with search string highlighted in yellow. If you use the replacement string function, the html report will also include the location information for the new SAS files. The location information in the html report is created with hyperlinks, so the user can directly open the files from the report.

INTRODUCTION

Sometimes, we want to change some codes in the SAS programs, like a variable name has been changed and we need to update the programs. If we have a lot of SAS files and we don't know which files contain the string we are looking for, it is time consuming to open each file, do the search and change. It will be nice if we have a tool available that can automatically do the work no matter how many SAS files we have. The macro presenting in this paper can do the search and replacement of any string in the SAS programs and create a user-friendly html report to access the information. **Error! Reference source not found.** shows you a sample html report generated by the macro, the search string in this example is "%macro" and the search folders are "C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample" and its subfolders.

The SAS System

location	sas_codes_line_no	sascodes
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	37	**%macro** print_sudoku(dsn);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	87	**%macro** store_initial_values;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	102	**%macro** solve;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	150	**%macro** convert_to_dense(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	186	**%macro** print_piday(dsn);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	283	**%macro** cdata;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	297	**%macro** cons_row(r);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	304	**%macro** cons_col(c);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	313	**%macro** cons_region(vars);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	320	**%macro** pds(solns=allsolns,varsel=MINR,maxt=900);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	417	**%macro** pds_out;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	440	**%macro** magic(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	491	**%macro** convert_to_dense(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	508	**%macro** print_msq(dsn);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp10.sas	237	**%macro** colorIdex(res_var=, proj=, palette=, out=);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp10.sas	265	**%macro** fnLegend(tfact=1.75,h=10,xStart=5,rhs=100,nCol=,nRow=,
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp10.sas	310	**%macro** setPatterns(map);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp11.sas	76	**%macro** colorIdex(res_var=, proj=, palette=, out=);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp11.sas	103	**%macro** fnLegend(tfact=1.75,h=10,xStart=5,rhs=100,nCol=,nRow=,
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp11.sas	148	**%macro** setPatterns(map);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp12.sas	25	**%macro** patterns();
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp12.sas	130	**%macro** pattern_sets();

Figure 1. The Screenshot of a Sample HTML Report for the Search Function

If we pass the replacement string to the macro, the macro will create new files for those files that contain the search string, replace the search string with the replacement string in those new files. The new files' names will have the prefix "n_" followed by the original file names. The new files will be saved in the same folder as the original files. Figure 2 shows you a sample html report for the replacement function. The search string is "%macro", the replacement string is "*This is a macro.;%macro" and the search folders are "C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample" and its subfolders.

The SAS System

location	New File Location	sas_codes_line_no	sascodes
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	37	%macro print_sudoku(dsn);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	87	%macro store_initial_values;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	102	%macro solve;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	150	%macro convert_to_dense(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	186	%macro print_piday(dsn);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	283	%macro cdata;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	297	%macro cons_row(r);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	304	%macro cons_col(c);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	313	%macro cons_region(vars);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	320	%macro pds(solns=allsolns,varsel=minr,maxt=900);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	417	%macro pds_out;
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	440	%macro magic(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	491	%macro convert_to_dense(n);
C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\clp1.sas	C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample\n_clp1.sas	508	%macro print_msq(dsn);

Figure 2. The Screenshot of a Sample HTML Report for the Replacement Function

The user can click hyperlinks in the report to directly open the files.

THE MACRO PARAMETERS

You can find the whole codes of the macro at the end of the paper. Below is the layout of the macro:

```
%macro SearchReplace(foldernm=,searchstring=,replacestring=%str(),htmldir=
%str(c:\result.html));
/*the codes are at the end of the paper*/
%mend;
```

- The "foldernm" is used to indicate the name of the search folder.
- The "searchstring" is used to save the search string.

- The "replacestring" is used to save the replacement string. If you don't want to use the replacement function, you don't need to pass the value to this macro variable.

- The "htmldir" is used to indicate where you want to save the html report. If you don't pass a value to this macro variable, by default, the html report will be saved as "c:\result.html".

CALL THE MACRO

The following macro call will search the "%macro" in all the SAS programs located in the "C:\Program Files\SASHome\x86\SASFoundation\9.4\or\sample" folder and its subfolders. If the macro finds any SAS files that contain the search string, the macro will replace the search string with the replacement string "*this is a macro;%macro". Then the macro will create new SAS files. The new SAS files are named as "n_" followed by the original file names, for e.g, if the original file name is "test. sas", the new file name will be "n_test.sas". The new files will be saved in the same folder as the original files. The html report will be saved as the "c:\result1.html".

```
%SearchReplace(foldernm=%str(C:\Program
Files\SASHome\x86\SASFoundation\9.4\or\sample),
searchstring=%str('%macro'),replacestring=%str('*this is a macro;%macro'),
htmldir=%str(c:\result1.html));
```

If you just want to use the search function, you don't need to pass a value to the "replacestring" macro variable, the following SAS codes show you an example, by calling the macro in this way, it will search the string "%macro" in the "C:\test" folder and its subfolders, the html report will be saved as "c:\result.html" by default:

```
%SearchReplace(foldernm=%str(C:\test),searchstring=%str('%macro'));
```

THE MACRO CODES

Below are all the sas codes for macro SearchReplace:

```
%macro SearchReplace
(foldernm=,searchstring=,replacestring=%str(),htmldir=%str(c:\result.html));
option mprint mlogic symbolgen NOXWAIT NOXSYNC;

filename ren pipe "dir ""&foldernm.\*.sas"" /b /s";
%put &foldernm.;

*dirinfo is a SAS data set that saves all the file information for the searching
folder;
data dirinfo;
infile ren pad;
input wholename $250.;
format filename $250.;
filename=cats(scan(scan(wholename,-1,'\'),1,'.'));
run;

data _null_;
set dirinfo end=end;
num=cats(_n_);
call symput("m"||num,cats(wholename));
call symput("n"||num,cats(filename));
if end then call symput("file_ct",num);
run;

%do i=1 %to &file_ct.;
```

```
*sasfile_&i. is the SAS data set that saves the SAS program codes;
data sasfile_&i.;
infile "&&m&i"
delimiter = '@@' missover dsd lrecl=32767 firstobs=1 TERMSTR=CRLF;
informat all $char5000. ;
input   all $ ;
run;

data sasfile_&i.;
length wholename $200. filename $50.;
wholename="&&m&i";
filename="&&n&i";
set sasfile_&i.;
line_no=_n_;
run;

*contain_string_sasfile_&i. contains the SAS codes that have the search string;
data contain_string_sasfile_&i.;
set sasfile_&i.;
if index(all,%str(&searchstring.)) >0;
run;
%end;

*search_result is the SAS data set that contains all the search_results;
data search_result;
set contain_string_sasfile_:;
run;

proc sort data=search_result;by wholename line_no;run;

*use this proc sql to check if the search_result data set is empty or not;
proc sql noprint;
select count(*) into :search_row_ct from search_result;
quit;

*htmlresult the data set that contains the data to be printed to the html
report;
data htmlresult;
length location $300;
%if &replacestring. ^= and &search_row_ct. ^= 0 %then %do;
length newloc $300;
%end;
set search_result;
location=cats("<a href='",wholename,"'>",wholename,"</a><br>");
%if &replacestring. ^= and &search_row_ct. ^= 0 %then %do;
newloc=cats("<a
href='",tranwrd(wholename,cats(filename,".sas"),""),cats("n_",filename,".sas"),"
'>",tranwrd(wholename,cats(filename,".sas"),""),cats("n_",filename,".sas"),"</a>
<br>");
label newloc="New File Location";
%end;
sascodes=tranwrd(all,&searchstring.,catx(" ","^S={background=yellow
font_weight=bold}",&searchstring.,"^S={}"));
run;

data htmlresult;
set htmlresult;
```

```
rename line_no=sas_codes_line_no;
drop all wholename filename;
run;

*print the result to the html report;
ods html file="&htmldir.";

ods escapechar='^';
%if  &search_row_ct. ^= 0 %then %do;
proc print data=htmlresult noobs label;run;
%end;
/*if didn't find any files contain the search string, the following message will
print to the html report;*/
%else %do;
ods html text="<H1 align='center'>Don't find &searchstring. in the SAS
files.</H1>";
%end;
ods html close;

*use the internet explorer to open the html report;
x "start iexplore &htmldir.";

*do the replacement function;
%if &replacestring. ^= and &search_row_ct. ^= 0 %then %do;
*the all_sasfiles is the SAS data set that has all the SAS codes in all the SAS
programs;
data all_sasfiles;
set sasfile_:;
run;

* the new_result data set contains the new SAS codes that contain the search
string and
have been replaced by the replacement string;
data new_result;
set search_result;
newcodes=tranwrd(all,&searchstring.,&replacestring.);
run;

/*the following codes merge the replaced SAS codes to the all SAS codes file,
the data set
new_sas_files contains all the SAS codes with the newly replaced SAS codes for
those files that have
the search string;*/
proc sql;
create table sasfiles_contain_string as
select * from all_sasfiles
where wholename in (select distinct wholename from search_result)
order by wholename,line_no;

create table new_sas_files as
select s.*,newcodes
from sasfiles_contain_string as s left join new_result as n
on s.wholename=n.wholename and s.line_no=n.line_no;
quit;

data new_sas_files;
set new_sas_files;
```

```
if newcodes="" then newcodes=all;
run;

*the files data set contains the original file location and the new file
location information for those files
that have the search string;
proc sql;
create table files as
select distinct
wholename,cats(tranwrd(wholename,cats(filename,".sas"),""),cats("n_",filename,".
sas")) as newloc
from new_sas_files;
quit;

data _null_;
set files end=end;
num=cats(_n_);
call symput("filenm"||num,cats(wholename));
call symput("loc"||num,cats(newloc));
if end then call symput("ct",num);
run;

*create new SAS files for those files that have the search string and replace
the search string
with the replacement string;
%do i=1 %to &ct.;
data f&i.;
set new_sas_files;
where wholename="&&filenm&i";
keep newcodes;
run;

data _null_;
set f&i.;
file "&&loc&i." notitles noprint;
put newcodes;
run;

proc sql;drop table f&i.;quit;
%end;
%end;
proc datasets lib=work kill;run;quit;
%MEND;
```

CONCLUSION

The macro presented in this paper can be used as a helpful tool to search and replace any strings in the SAS program files. Also you can further extend the functions of the macro by searching strings with different patterns, for e.g, search for all the string that start with the same prefix, suffix or middle parts, you can refer to this paper [1] for more details about how to search for different patterns.

ACKNOWLEDGMENTS

REFERENCES

1. Ting Sa, Yanhong Liu. A Simple Macro to Select Various Variables Lists. The Pharmaceutical Industry SAS® Users Group Proceedings 2015, May17-20, 2015, Orlando, USA.

CONTACT INFORMATION

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

Ting Sa
Division of Biostatistics and Epidemiology, Cincinnati Children's Hospital Medical Center
5136363674
Ting.Sa@CCHMC.ORG

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.