/\*Create graphs for impact analysis at facility level\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Macro Name: corrgraph

\* SAS version: PC SAS9.4 \*

\* Function: create scatter plot with correlation coefficient displayed \*

\* Part of codes is edited from UCLA: Statistical Consulting Group http://www.ats.ucla.edu/stat/sas/faq/corrgraph.htm. \*

\* %corrgraph(file, text1, text2, ver1, ver2, index1, index2, zscore1, zscore2, label1, label2); \*

\* file---the dataset that has all the information for creating overall impact analysis and correlation graphs (including both model versions) \*

\* text1/text2---they are values of model variable in overall impact analysis table. e.g. "ECRI v17 POA (No Char)"/"ECRI v18 POA (No Char)" for ECRI \*

\* ver1/ver2---model version number, e.g. ERCI17/ECRI18 for ECRIC, which depends on what they appear in the dataset \*

\* ver\_norm1/ver\_norm2---model version number for normalized expected values, e.g. ERCI17\_RENORM\_EXPECTED/ECRI18\_RENORM\_EXPECTED for ECRIC, which depends on what they \*

\* appear in the dataset. For databrdige impact anlaysis these values can be as same as ver1/ver2 \*

\* index1/index2---index version for creating index by facility correlation graph, e.g. ECRI\_index\_17/ECRI\_index\_18 for ECRI \*

\* zscore1/zscore2---score version for creating zscore by facility correlation graph, e.g. ECRI\_zscore\_17/ECRI\_zscore\_18 for ECRI \*

\* label1/label2---label1 for index1/zscore1 in the graph. it is the label for x axis, i.e. how the axis x in the graph looks like. e.g. "ECRI v17 POA" for x-axis in ECRI\*

\* label2 for index2/zscore2 in the graph. it is the label for y axis, i.e. how the axis y in the graph looks like. e.g. "ECRI v18 POA" for y-axis in ECRI\*

\*Below is the example of applying the macro to ECRI impact analysis \*

\*%corrgraph(final\_ecri\_rami, "ECRI v17 POA (No Char)", "ECRI v18 POA (No Char)", ECRI17, ECRI18, ECRI\_index\_17, ECRI\_index\_18, ECRI\_zscore\_17, ECRI\_zscore\_18, "ECRI v17 POA", \*

\* "ECRI v18 POA"); \*

\*Note---the macro was developed based on the ERIC/RAMI/PFD impact analysis. Some minor changes might be needed if applied to differenct models

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*Libname directory is subject to change due to a temporary space it is ;

libname output "Y:\LynnL\Impact\OPRD\_2016\_11\analysis";

**Assigns a library named output for the location mentioned.**

%MACRO corrgraph(file, text1, text2, ver1, ver\_norm1, ver2, ver\_norm2, index1, index2, zscore1, zscore2, label1, label2);

**Creates a new macro corrgraph which takes input parameters as file, text1, text2, ver1, ver\_norm1, ver2, ver\_norm2, index1, index2, zscore1, zscore2, label1 and label2.**

Proc sql;

create table overall\_impact1 as

select &text1 as Model, sum(&ver1.\_qual\_discharges) as N, SUM(&ver1.\_observed)/sum(&ver1.\_qual\_discharges) as Avg\_observed,

SUM(&ver\_norm1.\_expected)/sum(&ver1.\_qual\_discharges) as Avg\_expected, SUM(&ver1.\_observed)/sum(&ver\_norm1.\_expected) as index

from output.&file;

quit;

**Creates table overall\_impact1 from the file macro variable in the output library and selects text1 as Model, summation of &ver1.\_qual\_discharges as N and summation of &ver1.\_observed divided by summation of &ver1.\_qual\_discharges as Avg\_observed and summation of &ver\_norm1.\_expected divided by summation of &ver1.\_qual\_discharges as Avg\_expected and summation of &ver1.\_observed divided by summation of &ver\_norm1.\_expected as index.**

proc sql;

create table overall\_impact2 as

select &text2 as Model, sum(&ver2.\_qual\_discharges) as N, SUM(&ver2.\_observed)/sum(&ver2.\_qual\_discharges) as Avg\_observed,

SUM(&ver\_norm2.\_expected)/sum(&ver2.\_qual\_discharges) as Avg\_expected, SUM(&ver2.\_observed)/sum(&ver\_norm2.\_expected) as index

from output.&file;

quit;

**Creates table overall\_impact2 from the file macro variable in the output library and selects text2 macro variable as Model, summation of &ver2.\_qual\_discharges as N and summation of &ver2.\_observed divided by summation of &ver2.\_qual\_discharges as Avg\_observed and summation of &ver\_norm2.\_expected divided by summation of &ver2.\_qual\_discharges as Avg\_expected and summation of &ver2.\_observed divided by summation of &ver\_norm2.\_expected as index.**

data overall\_impact;

set overall\_impact1 overall\_impact2;

run;

**Creates a dataset overall\_impact by combining both overall\_impact1 and Overall\_impact2.**

data overall\_impact\_final(drop=lag\_index);

set overall\_impact;

lag\_index=lag(index);

Index\_diff=index-lag\_index;

index\_per\_diff=(index-lag\_index)/lag\_index;

run;

**Creates a dataset overall\_impact\_final and it will have two new additional columns. The first one is index\_diff which is the difference between current index variable value and previous observations index variable value and the second column will be the index\_per\_diff which is calculated by ratio of index\_diff and index value of the previous observation.**

proc corr data=output.&file noprint outp=\_rcorr\_index;

var &index1 &index2;

run;

**The above correlation procedure gives the strength of the linear relationship between two macro variables index1 and index2 which are passed in the macro at the beginning and the out is a table \_rcorr\_index with the correlation strength values.**

proc print data=\_rcorr\_index;run;

**Prints the correlation table created above.**

\*Put correlation coefficient to r;

data \_null\_;

set \_rcorr\_index;

if \_n\_=4 then do;

r=floor(&index2\*1000)/1000+0;

if r = 1 then r1 = 1;

else if r = -1 then r1 = -1;

else if r = 0 then r1 = 0;

else r1 = input(r, 20.);

call symput('r1',strip(r1));

end;

run;

**The datastep is used to create a new macro variable ‘r1’ and it is created by below procedure:**

**Reads the dataset \_rcorr\_index and reads only the fourth observation in the dataset. For fourth observation it would calculate r with formula as floor(&index2\*1000)/1000+0 the floor function will just rounds off the value to the lowest integer. Suppose if the result is 0.5 then floor(0.5) is 0. Then after calculation of r it would be checked if it is 1 then r1 would be 1 if r=-1 then r1 is -1 and if r=0 then r1 is zero if all the above conditions doesn’t satisfy then r1 would be equal to r.**

proc corr data=output.&file noprint outp=\_rcorr\_zscore ;

var &zscore1 &zscore2;

**run;**

**The above correlation procedure gives the strength of the linear relationship between two macro variables zscore1 and zscore2 from dataset name present in macro variable file in the output library which are passed in the macro at the beginning and the output is a table \_rcorr\_zscore with the correlation strength values.**

data \_null\_;

set \_rcorr\_zscore;

if \_n\_=4 then do;

r=floor(&zscore2\*1000)/1000+0;

if r = 1 then r2 = 1;

else if r = -1 then r2 = -1;

else if r = 0 then r2 = 0;

else r2 = input(r, 20.);

call symput('r2',strip(r2));

end;

run;

**The datastep is used to create a new macro variable ‘r2’ and it is created by below procedure:**

**Reads the dataset \_rcorr\_zscore and reads only the fourth observation in the dataset. For fourth observation it would calculate r with formula as floor(&zscore2\*1000)/1000+0 the floor function will just rounds off the value to the lowest integer. Then after calculation of r it would be checked if it is 1 then r2 would be 1 if r=-1 then r2 is -1 and if r=0 then r2 is zero if all the above conditions doesn’t satisfy then r2 would be equal to r.**

%put &r1 &r2;

**Prints the values of r1 and r2 in the log.**

ods excel file="Y:\LynnL\Impact\OPRD\_2016\_11\analysis\&ver1.\_&ver2.\_correlation.xlsx" options(sheet\_interval='none' );

**ODS statement mentions that whatever that will be printed will be written into a new excel file with name as &ver1.\_&ver2.\_correlation.xlsx where ver1 and ver2 are the macro variables which are passed at the beginning instead of writing into the standard output window in SAS.**

ods excel options(start\_at='A1' sheet\_name='Scatter Plot Correlation' embedded\_titles='yes');

**Content will be written in above mentioned excel workbook and the sheet name would be 'Scatter Plot Correlation'.**

title;

Title1 j=left italic "Overall Statistics";

**Prints title in the excel sheet.**

proc print data=overall\_impact\_final noobs;

var Model N avg\_observed avg\_expected index Index\_diff index\_per\_diff;

format N comma10. avg\_observed 6.4 avg\_expected 6.4 index 6.4 Index\_diff 6.4 index\_per\_diff percent7.2;

run;

**Prints data from overall\_impact\_final dataset and the columns which will be printed are Model N avg\_observed avg\_expected index Index\_diff index\_per\_diff. Various columns will be formatted as per the format statement.**

ods excel;

proc sql ;

create table range1 as

select ceil(min(&index2)-1) as min1, ceil(max(&index2)+1) as max1, int((ceil(max(&index2)+1)- ceil(min(&index2)-1))/4) as num1,

ceil(min(&index1)-1) as min2, ceil(max(&index1)+1) as max2, int((ceil(max(&index1)+1)- ceil(min(&index1)-1))/4) as num2

from output.&file;

quit;

**Creates a table range1 from file macro variable in the output library by selecting columns as per the values in the macro variables index2, index1. For example the output table has min1 which is derived as below:**

**Minimum of column name in the macro variable index2 substract 1 from the result and then take ceil means lowest integer of the result. So if the index2 macro variable has value as “Column1” then equivalent function will be ceil(min(column1)-1).**

**Similarly with max1 and other columns created.**

proc sql noprint;

select min(min1, min2), max(max1, max2), max(num1,num2)

into :min,:max,:num

from range1;

quit;

**The above SQL creates 3 new macro variables those are min, max and num. min variable is created by taking the minimum of the two variables that is min1 and min2 and max variable is created by taking maximum of two variables max1 and max2 and num is created by taking the max of num1 and num2 from the table range1.**

%put &min &max &num;

**Prints min, max and num macro variable values in the log.**

\*Create the annotate dataset;

data anno;

function='move';

xsys='1'; ysys='1';

x=0; y=0;

output;

function='draw';

xsys='1'; ysys='1';

color='green';

x=100; y=100;

output;

run;

**Creates a dataset named anno which has only two observations with 5 variables. Function, x, y, xsys and ysys with values as assigned in the data step.**

proc gplot data=output.&file;

label &index2=&label2 &index1=&label1;

plot &index2\*&index1/anno=anno vaxis=axis1 haxis=axis2 grid frame;

axis1 order=(0 to &max by &num &max);

axis2 order=(0 to &max by &num &max);

title1 j=center h=2 bold "Index by Hospital";

title2 j=center h=1.5 bold "(Correlation Coefficient=&r1)";

run;

quit;

**Creates a scatter plot from table which is present in the macro variable named file and from output library. The horizontal axis has index2 variable values and vertical axis has index1 values.**

ods excel;

proc sql;

create table range2 as

select ceil(min(&zscore2)-1) as min1, ceil(max(&zscore2)+1) as max1, int((ceil(max(&zscore2)+1)- ceil(min(&zscore2)-1))/4) as num1,

ceil(min(&zscore1)-1) as min2, ceil(max(&zscore1)+1) as max2, int((ceil(max(&zscore1)+1)- ceil(min(&zscore1)-1))/4) as num2

from output.&file;

quit;

**Creates a table range2 from file macro variable in the output library by selecting columns as per the values in the macro variables zscore2, zscore1. For example the output table has min1 which is derived as below:**

**Minimum of column name in the macro variable zscore2 substract 1 from the result and then take ceil means lowest integer of the result. So if the zscore2 macro variable has value as “Column1” then equivalent function will be ceil(min(column1)-1).**

proc sql noprint;

select min(min1, min2), max(max1, max2), max(num1,num2)

into :min,:max,:num

from range2;

quit;

**The above SQL creates 3 new macro variables those are min, max and num. min variable is created by taking the minimum of the two variables that is min1 and min2 and max variable is created by taking maximum of two variables max1 and max2 and num is created by taking the max of num1 and num2 from the table range2.**

proc gplot data=output.&file;

label &zscore2=&label2 &zscore1=&label1;

plot &zscore2\*&zscore1/anno=anno vaxis=axis1 haxis=axis2 regeqn frame grid;

axis1 order=(&min to &max by &num &max);

axis2 order=(&min to &max by &num &max);

title1 j=center h=2 bold "Z-Score by Hospital";

title2 j=center h=1.5 bold "(Correlation Coefficient=&r2)";

run;

quit;

ods excel close;

**Creates a scatter plot from table which is present in the macro variable named file and from output library. The horizontal axis has zscore2 variable values and vertical axis has zscore1 values.**

%mend;

%corrgraph(final\_oprd\_cst, "OPRD-CST v2016", "OPRD-CST v2017", CST2016, CST2016\_RENORM, CST2017, CST2017\_RENORM, CST\_index\_2016, CST\_index\_2017, CST\_zscore2016, CST\_zscore2017, "CST v2016", "CST v2017");

%corrgraph(final\_oprd\_chg, "OPRD-CHG v2016", "OPRD-CHG v2017", CHG2016, CHG2016\_RENORM, CHG2017, CHG2017\_RENORM, CHG\_index\_2016, CHG\_index\_2017, CHG\_zscore2016, CHG\_zscore2017, "CHG v2016", "CHG v2017");

**The above two statements will call the macro with parameters.**