/\*\*\*\* WoE and SFA \*\*\*\*/

\*Creating a good bad label in the data;

data EM\_LIB.EM\_Qual\_MVI\_FINAL;

set EM\_LIB.EM\_Qual\_MVI\_FINAL;

if INTODEF = 0 then good\_bad\_label = "GOOD";

if INTODEF = 1 then good\_bad\_label = "BAD";

Run;

proc sql;

create table woe\_sum\_check as select "AVAIL135" as Var\_Name ,AVAIL135 as Responses ,count(\*) as freq, sum(intodef) as def\_count from

EM\_LIB.EM\_Qual\_MVI\_FINAL

group by AVAIL135;

quit;

data woe\_sum\_check;

set woe\_sum\_check;

bad = def\_count;

good = freq - def\_count;

bad\_prct=bad/sum(bad);

run;

proc sql;

create table woe\_sumry\_AVAIL135 as select Var\_Name,Responses, bad as bad,good as good ,bad/sum(bad) as bad\_prct, good/sum(good) as good\_prct from

woe\_sum\_check;

quit;

options symbolgen mprint mlogic;

\*Creating a dataset to store IV values;

Proc sql;

create table WOE\_Quals\_MEP\_EM (

Variable char(100),

bad num,

good num,

bad\_prct num,

good\_prct num,

woe num,

IV num);

quit;

data WoE\_Summary\_all\_quals;

set \_NULL\_;

run;

%let final\_qual\_vars =

AVAIL135

BARRI026

BUSIN452

CAPIT050

COMPE618

ENVIR960

FISCA988

INDUS038

INDUS404

INDUS712

INDUS844

INFLA869

LABOU080

PRODU265

REGUL789

TECHN323

;

%Macro WOE\_Transformation ();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table woe\_sum\_&variable. as select "&variable." as Var\_Name, &variable. as Responses ,count(\*) as freq, sum(intodef) as def\_count from

EM\_LIB.EM\_Qual\_MVI\_FINAL

group by &variable.;

quit;

data woe\_sum\_&variable.;

set woe\_sum\_&variable.;

bad = def\_count;

good = freq - def\_count;

run;

proc sql;

create table woe\_sumry\_&variable. as select Var\_Name,Responses, bad as bad,good as good ,bad/sum(bad) as bad\_prct, good/sum(good) as good\_prct from

woe\_sum\_&variable.;

quit;

data woe\_sumry\_&variable.;

set woe\_sumry\_&variable.;

woe = log(bad\_prct/good\_prct);

iv = (bad\_prct-good\_prct)\*woe;

run;

proc transpose data=woe\_sumry\_&variable. out = woe\_sumry\_t\_&variable. name= &variable. prefix= Res\_;

run;

data woe\_sumry\_t\_&variable. (keep= &variable. sum\_&variable.);

set woe\_sumry\_t\_&variable.;

sum\_&variable. = sum(of Res\_:);

run;

proc transpose data= woe\_sumry\_t\_&variable. out= woe\_sumry\_tt\_&variable. Name= Variable;

id &variable.;

run;

data WoE\_Summary\_all\_quals;

set WoE\_Summary\_all\_quals woe\_sumry\_&variable.;

run;

data WOE\_Quals\_MEP\_EM;

set WOE\_Quals\_MEP\_EM woe\_sumry\_tt\_&variable.;

run;

%end;

%Mend;

%WOE\_Transformation()

data EM\_Qual\_MVI\_FINAL\_WoE;

set EM\_LIB.EM\_Qual\_MVI\_FINAL;

run;

data EM\_LIB.WoE\_Summary\_all\_quals\_EM (keep= Var\_Name Responses woe) ;

set WoE\_Summary\_all\_quals;

run;

/\*\*MERGE THE WOE FIGURES TO THE MAIN DATA SET\*\*/

%macro woe\_merge();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

data woe\_table\_&variable.(keep=Responses woe rename=(woe=&variable.\_woe));

set EM\_LIB.WoE\_Summary\_all\_quals\_EM;

where Var\_Name = "&variable.";

run;

proc sql;

create table EM\_Qual\_MVI\_FINAL\_WoE as select a.\*,b.&variable.\_woe from EM\_Qual\_MVI\_FINAL\_WoE a

left join woe\_table\_&variable. b on a.&variable.=b.Responses;

quit;

%end;

%mend;

%woe\_merge();

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

/\*\*EM Qual\*//

Data EM\_Dev;

set em\_lib.em\_mep\_dev\_sample\_16 ;

run;

proc contents data= em\_lib.em\_mep\_dev\_sample\_16 ; run;

proc freq data=em\_lib.em\_mep\_dev\_sample\_16;tables intodef \* country\_use; where country\_use="MU";run;

PROC SORT DATA = APL.GRADE\_RA5\_QUALITATIVES\_20191008 OUT = GRADE\_RA5\_QUAL\_14012020\_NODUPS NODUPKEY;

BY RA5\_SYSTEM CUSTOMERID ARCHIVEID CHARACTERISTIC\_ID;

RUN;

/\* 127,571,765 --> 64,936,049 \*/

PROC SQL;

SELECT

DISTINCT (Characteristic\_id) INTO :QUAL\_VAR\_MOD SEPARATED BY ' '

FROM EM\_LIB.GRADE\_RA5\_QUAL\_20191008\_NODUPS

WHERE Characteristic\_id NOT IN ('1UNS856', 'NON-C056', 'NON-C341', 'NON-C507')

;

QUIT;

proc contents data=

EM\_LIB.EM\_QUAL\_YR\_DEV ; run;

/\* EXTRACTING DATA FOR YR 2012 TO 2016 (ONLY RA5 DATA) FOR QUALITATIVE ANALYSIS \*/

DATA EM\_LIB.EM\_QUAL\_YR\_DEV;

SET EM\_Dev (WHERE = (YR > 2011 AND YR < 2017));

KEEP CUST\_ID

RELN\_ID

APPROVAL\_DATE

INTODEF

RA5\_SYSTEM

CARM\_INSTANCE

YR

CUSTOMERID

ARCHIVEID

COUNTRY\_USE

REP\_DATE

STMTDATE

DATE\_OF\_INCORPORATION

AUDITMETHOD

PD

log\_PD

SIC\_CODE\_DESCRIPTION

sample\_split

OOT\_Flag

'1UNS856'N

'NON-C056'N

'NON-C341'N

'NON-C507'N

&QUAL\_VAR\_MOD.

;

RUN;

/\* 25,738 \*/

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

Quick check on GENER047 and GENER662 the two coomon questions whose repsonses are also same.

It's found that GENER662 is mainly for HBAP LC Scorecard ; where as GENER047 is for HBAP\_MIDDLE\_MARKET Scorecard.

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

data qual\_check;

set EM\_Dev;

KEEP CUST\_ID

RELN\_ID

APPROVAL\_DATE

INTODEF

RA5\_SYSTEM

CARM\_INSTANCE

YR

CUSTOMERID

ARCHIVEID

COUNTRY\_USE

REP\_DATE

STMTDATE

DATE\_OF\_INCORPORATION

AUDITMETHOD

PD

log\_PD

SIC\_CODE\_DESCRIPTION

sample\_split

OOT\_Flag

GENER047

GENER662

SCORECARD\_NAME

;

RUN;

proc freq data=qual\_check; tables SCORECARD\_NAME\*GENER047;run;

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

Macro and code to determine the number of missing values for the qualitatives

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

data char\_miss\_check;

set \_NULL\_;

run;

%let qual\_variables =

&QUAL\_VAR\_MOD.

;

%MACRO char\_miss\_cnt();

%do j=1 %to %sysfunc(countw(&qual\_variables.));

%let Factor=%scan(&qual\_variables.,&j.);

proc sql;

create table test\_var\_miss as

select "&Factor" as Variable ,

(select DISTINCT count(\*) as miss\_cnt from EM\_LIB.EM\_QUAL\_YR\_DEV where &Factor. =' ') as missing\_obs,

count(\*) as total from EM\_LIB.EM\_QUAL\_YR\_DEV;

quit;

data char\_miss\_check;

set char\_miss\_check test\_var\_miss;

run;

%end;

%mend;

%char\_miss\_cnt();

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

List of characteristics\_id and description :

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

PROC SQL;

SELECT

DISTINCT characteristic\_id

,Description

FROM EM\_LIB.GRADE\_RA5\_QUAL\_20191008\_NODUPS

;

QUIT;

PROC SQL;

SELECT

DISTINCT characteristic\_id

,Description

FROM EM\_LIB.GRADE\_RA5\_QUAL\_20191008\_NODUPS where characteristic\_id in

('AVAIL135', 'BARRI026', 'BUSIN452', 'CAPIT050', 'COMPE618', 'ENVIR960', 'FISCA988', 'INDUS038', 'INDUS404', 'INDUS712', 'INDUS844', 'INFLA869', 'LABOU080', 'PRODU265', 'REGUL789', 'TECHN323')

;

QUIT;

%let final\_qual\_vars =

AVAIL135

BARRI026

BUSIN452

CAPIT050

COMPE618

ENVIR960

FISCA988

INDUS038

INDUS404

INDUS712

INDUS844

INFLA869

LABOU080

PRODU265

REGUL789

TECHN323

;

/\*\*\* EP Suggested Qual variables : \*\*\*/

%let ep\_quals =

CYCLI202

AVAIL480

EFFEC028

INTRM001

GENER047

GENER662

CAPIT001

REFIN128

REFIN202

ACCES165

ACCES246

CHARA002

CONCE002

;

/\*\*\*\*\* get the response of each questions \*\*\*\*\*\*\*\*\*\*\*\*\*\*/

proc freq data=EM\_LIB.EM\_QUAL\_YR\_DEV ; table &final\_qual\_vars.;run;

proc freq data=em\_qual\_2;table &final\_qual\_vars.;run;

DATA EM\_LIB.EM\_QUAL\_YR\_DEV\_FINAL;

SET EM\_LIB.EM\_QUAL\_YR\_DEV;

KEEP CUST\_ID

RELN\_ID

APPROVAL\_DATE

INTODEF

RA5\_SYSTEM

CARM\_INSTANCE

YR

CUSTOMERID

ARCHIVEID

COUNTRY\_USE

REP\_DATE

STMTDATE

DATE\_OF\_INCORPORATION

AUDITMETHOD

PD

log\_PD

SIC\_CODE\_DESCRIPTION

sample\_split

OOT\_Flag

&final\_qual\_vars.

/\*&ep\_quals.\*/

;

RUN;

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

TO CHECK DEFAULT RATE ACCROSS DIFFERENT BUCKETS OF THE REPONSES FOR THE QUALITATIVE VARIABLES

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

data DR\_Base;

set \_null\_;

run;

%macro DR\_analysis();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table Def\_rate as

select distinct &variable. as responses , "&variable." as Var\_Name,

sum(intodef)/count(\*) as Default\_Rate,

count(\*) / 25738 as Avilability

from EM\_LIB.EM\_QUAL\_YR\_DEV group by &variable.;

quit;

data DR\_Base;

set DR\_Base Def\_Rate;

run;

%end;

%mend;

%DR\_analysis()

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

CHECK Log\_PD distribution ACCROSS THE IMPUTED RESPONSES WHETHER IT IS MONOTONIC NOW OR NOT :

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

data LogPD\_Base;

set \_null\_;

run;

%macro LogPD\_analysis();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table logPD\_resp\_dist as

select distinct &variable. as responses , "&variable." as Var\_Name,

mean(PD) as mean\_PD,

count(\*) / 25738 as Avilability

from EM\_LIB.EM\_QUAL\_YR\_DEV group by &variable.;

quit;

data LogPD\_Base;

set LogPD\_Base logPD\_resp\_dist;

run;

%end;

%mend;

%LogPD\_analysis()

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

TO CHECK DEFAULT RATE ACCROSS DIFFERENT BUCKETS OF THE REPONSES FOR THE EP SUGGESTED QUALITATIVE VARIABLES

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

PROC SQL;

SELECT

DISTINCT characteristic\_id

,Description

FROM EM\_LIB.GRADE\_RA5\_QUAL\_20191008\_NODUPS where characteristic\_id in

('CYCLI202', 'AVAIL480', 'EFFEC028', 'INTRM001', 'GENER047', 'GENER662', 'CAPIT001', 'REFIN128', 'REFIN202', 'ACCES165', 'ACCES246', 'CHARA002', 'CONCE002')

;

QUIT;

data DR\_Base\_ep\_qual;

set \_null\_;

run;

%macro DR\_analysis\_ep\_qual();

%do j=1 %to %sysfunc(countw(&ep\_quals.));

%let variable=%scan(&ep\_quals.,&j.);

proc sql;

create table Def\_rate\_ep as

select distinct &variable. as responses , "&variable." as Var\_Name,

sum(intodef)/count(\*) as Def\_Rate ,

count(\*) / 25738 as Avilability

from EM\_LIB.EM\_QUAL\_YR\_DEV group by &variable.;

quit;

data DR\_Base\_ep\_qual;

set DR\_Base\_ep\_qual Def\_rate\_ep;

run;

%end;

%mend;

%DR\_analysis\_ep\_qual()

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

CONVERTING THE CHARACTERISTICV RESPONSES OF THE QUAL VARIABLES TO NUMERICAL NUMBERS (0--> FOR LESS CRR 3/4--> FOR HIGH CRR)

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

data em\_qual\_1;

SET EM\_LIB.EM\_QUAL\_YR\_DEV;

if AVAIL135 = 'NONE' then AVAIL135\_f=0 ; else if AVAIL135='FEW' then AVAIL135\_f=1 ; else if AVAIL135 in ('SOME','MANY')then AVAIL135\_f=2; /\*clubbed two options\*/

if BARRI026 = 'LOW' then BARRI026\_f=0; else if BARRI026 in ('MODERATE','HIGH') then BARRI026\_f=1 ; /\*clubbed two options High & Moderate to make monotonic\*/

if BUSIN452 = 'LOW' then BUSIN452\_f=0; else if BUSIN452 in ('MODERATE','HIGH') then BUSIN452\_f=1; /\*clubbed two options High & Moderate to make monotonic\*/

if CAPIT050= 'LOW' then CAPIT050\_f=0 ;else if CAPIT050 in ('MODERATE','HIGH','VERY HIGH') then CAPIT050\_f=1; /\*clubbed three options Very High , High & Moderate to make monotonic\*/

if COMPE618 in ('NONE' ,'UNTHREATENING') then COMPE618\_f=0 ;else if COMPE618='HOSTILE' then COMPE618\_f=1; else if COMPE618='AGGRESSIVE' then COMPE618\_f=2; /\*clubbed two options to make monotonic\*/

if ENVIR960 = 'LOW' then ENVIR960\_f=0 ; else if ENVIR960 in ('HIGH','MODERATE') then ENVIR960\_f=1; /\*clubbed two options High & Moderate to make monotonic\*/

if FISCA988 = 'HIGH' then FISCA988\_f=0; else if FISCA988='MODERATE' then FISCA988\_f=1 ; else if FISCA988='LOW' then FISCA988\_f=2;

if INDUS038 in ('STRONG INCREASE','INCREASE') then INDUS038\_f=0 ; else if INDUS038 ='STABLE' then INDUS038\_f=1 ; else if INDUS038 in ('DECREASE','STRONG DECREASE') then INDUS038\_f=2;

if INDUS404 ='GROWTH' then INDUS404\_f=0 ; else if INDUS404='STABLE/MATURE' then INDUS404\_f=1; else if INDUS404='DECLINE' then INDUS404\_f=3;

if INDUS712 in ('STRONG INCREASE' ,'INCREASE','STABLE') then INDUS712\_f=0 ; if INDUS712='DECREASE' then INDUS712\_f=1; /\*clubbed 3 options Low & Avg to make monotonic\*/

if INDUS844 in ('LOW' ,'AVERAGE') then INDUS844\_f=0 ; else if INDUS844 = 'HIGH' then INDUS844\_f=1; /\*clubbed two options Low & Avg to make monotonic\*/

if INFLA869 in ('LOW','MODERATE') then INFLA869\_f=0 ; else if INFLA869='HIGH' then INFLA869\_f=1; /\*clubbed two options Low & Avg to make monotonic\*/

if LABOU080 in ('LOW' ,'MODERATE') then LABOU080\_f=0 ; else if LABOU080='HIGH' then LABOU080\_f=1; /\*clubbed two options Low & Avg to make monotonic\*/

if PRODU265 in ('HIGH','CUSTOM') then PRODU265\_f=0; else if PRODU265='COMMODITY' then PRODU265\_f=1; else if PRODU265='SLIGHT' then PRODU265\_f=2; else if PRODU265='MODERATE' then PRODU265\_f=3; /\*clubbed two options make monotonic\*/

if REGUL789= 'LOW' then REGUL789\_f=0 ; else if REGUL789 in ('MODERATE','HIGH') then REGUL789\_f=1; /\*clubbed two options HIGH & MODERATE to make monotonic\*/

if TECHN323= 'INSIGNIFICANT' then TECHN323\_f=0; else if TECHN323= 'LOW' then TECHN323\_f=1; else if TECHN323= 'MODERATE' then TECHN323\_f=2; else if TECHN323='HIGH' then TECHN323\_f=2; else if TECHN323='VERY HIGH' then TECHN323\_f=2; /\*clubbed 3 options make monotonic\*/

/\*\*/

/\*/\*\*Variables suggested by EP \*\*/\*/

/\*if CYCLI202 = 'ANTI-CYCLICAL' then CYCLI202\_f=0 ;

/\*else if CYCLI202 = 'MODERATELY CYCLICAL' then CYCLI202\_f=1;\*/\*/

/\*\*/

/\*if GENER047 in ('VERY UNFAVOURABLE','UNFAVOURABLE','NONE') then GENER\_f=0;

/\*else if GENER047 in ('FAVOURABLE','MIXED','NEUTRAL','VERY FAVOURABLE') THEN GENER\_f=1;\*/\*/

/\*if GENER662 in ('VERY UNFAVOURABLE','UNFAVOURABLE','NONE') then GENER\_f=0;

/\*else if GENER662 in ('FAVOURABLE','MIXED','NEUTRAL','VERY FAVOURABLE') THEN GENER\_f=1;\*/\*/

/\*\*/

/\*if REFIN128 = 'HIGH' then REFIN\_f=2;

/\*else if REFIN128= 'MODERATE' then REFIN\_f=1; else if REFIN128 in ('LOW','IRRELEVANT') then REFIN\_f=2;\*/

/\*if REFIN202 = 'HIGH' then REFIN\_f=2; else if REFIN202= 'MODERATE' then REFIN\_f=1; else if REFIN202 = 'LOW' then REFIN\_f=2;\*/

/\*\*/

/\*if ACCES165 in ('>200 PERCENT') then ACCES\_f= 0 ; else if ACCES165 in ('>50-100 PERCENT','>100-200 PERCENT') then ACCES\_f=1; else if ACCES165 in ('NONE','0-10 PERCENT','>10-25 PERCENT','>25-50 PERCENT') then ACCES\_f=2;\*/

/\*if ACCES246 in ('>200 PERCENT') then ACCES\_f= 0 ; else if ACCES246 in ('>50-100 PERCENT','>100-200 PERCENT') then ACCES\_f=1; else if ACCES246 in ('NONE','0-10 PERCENT','>10-25 PERCENT','>25-50 PERCENT') then ACCES\_f=2;\*/

/\*\*/

/\*if CAPIT001 ='LOW' then CAPIT001\_f=0 ; else if CAPIT001 in ('HIGH','MODERATE') then CAPIT001\_f=1;\*/

/\*\*/

/\*if CONCE002 in ('More than 30% and Less than 50%', 'More than or equal to 50%') then CONCE002\_f=0; else if CONCE002 = 'Less than or equal to 30% / Not applicable' then CONCE002\_f=1;\*/

/\*\*/

run;

data EM\_LIB.em\_qual\_2;

set em\_qual\_1 ;

keep

CUST\_ID

RELN\_ID

APPROVAL\_DATE

INTODEF

RA5\_SYSTEM

CARM\_INSTANCE

YR

CUSTOMERID

ARCHIVEID

COUNTRY\_USE

REP\_DATE

STMTDATE

DATE\_OF\_INCORPORATION

AUDITMETHOD

PD

log\_PD

SIC\_CODE\_DESCRIPTION

sample\_split

AVAIL135\_f

BARRI026\_f

BUSIN452\_f

CAPIT050\_f

COMPE618\_f

ENVIR960\_f

FISCA988\_f

INDUS038\_f

INDUS404\_f

INDUS712\_f

INDUS844\_f

INFLA869\_f

LABOU080\_f

PRODU265\_f

REGUL789\_f

TECHN323\_f

;

rename

AVAIL135\_f = AVAIL135

BARRI026\_f = BARRI026

BUSIN452\_f = BUSIN452

CAPIT050\_f = CAPIT050

COMPE618\_f = COMPE618

ENVIR960\_f = ENVIR960

FISCA988\_f = FISCA988

INDUS038\_f = INDUS038

INDUS404\_f = INDUS404

INDUS712\_f = INDUS712

INDUS844\_f = INDUS844

INFLA869\_f = INFLA869

LABOU080\_f = LABOU080

PRODU265\_f = PRODU265

REGUL789\_f = REGUL789

TECHN323\_f = TECHN323

;

run;

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

Imputing missing values with mode grouped by YR and Country\_Use

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

data base;

set \_null\_;

run;

%macro descriptive\_mode();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc univariate data=EM\_LIB.em\_qual\_2 modes;

var &variable.;

class COUNTRY\_USE YR ; \*this line has been modified as per our requirement;

output out=&variable.1 mode=&variable.\_mode;

run;

data &variable.1;

length variable $50.;

set &variable.1;

variable="&variable.";

run;

data &variable.2(rename=&variable.\_mode = mode);

set &variable.1;

run;

data base;

set base &variable.2;

run;

%end;

%mend;

%descriptive\_mode();

/\*/\*This part will do the missing value imputation with the mode;\*/

%macro MVI\_QUAL();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table EM\_LIB.em\_qual\_2 as select a.\*,b.&variable.\_mode

from EM\_LIB.em\_qual\_2 as a left join &variable.1 as b

on a.country\_use = b.country\_use and a.YR=b.YR;

quit;

%end;

%mend;

%MVI\_QUAL();

/\*Macro for MVI of variables \*/

%macro MVT\_final();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let var=%scan(&final\_qual\_vars.,&j.);

Data EM\_LIB.em\_qual\_2(drop=&var.\_mode);

Set EM\_LIB.em\_qual\_2;

if &var. = . then do ;

&var. = &var.\_mode ;

end;

run;

%end;

%mend;

%MVT\_final();

data EM\_LIB.EM\_Qual\_MVI\_FINAL;

set EM\_LIB.em\_qual\_2;

run;

proc freq data=EM\_LIB.EM\_Qual\_MVI\_FINAL ; tables &final\_qual\_vars.; run;

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

CHECK DR ACCROSS THE IMPUTED RESPONSES WHETHER IT IS MONOTONIC NOW OR NOT :

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

data DR\_Base;

set \_null\_;

run;

%macro DR\_analysis();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table Def\_rate as

select distinct &variable. as responses , "&variable." as Var\_Name,

sum(intodef)/count(\*) as Default\_Rate,

count(\*) / 25738 as Avilability

from EM\_LIB.EM\_Qual\_MVI\_FINAL group by &variable.;

quit;

data DR\_Base;

set DR\_Base Def\_Rate;

run;

%end;

%mend;

%DR\_analysis()

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

CHECK Log\_PD distribution ACCROSS THE IMPUTED RESPONSES WHETHER IT IS MONOTONIC NOW OR NOT :

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

data LogPD\_Base;

set \_null\_;

run;

%macro LogPD\_analysis();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc sql;

create table logPD\_resp\_dist as

select distinct &variable. as responses , "&variable." as Var\_Name,

mean(log\_PD) as mean\_Log\_PD,

count(\*) / 25738 as Avilability

from EM\_LIB.EM\_Qual\_MVI\_FINAL group by &variable.;

quit;

data LogPD\_Base;

set LogPD\_Base logPD\_resp\_dist;

run;

%end;

%mend;

%LogPD\_analysis()

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

Code block to check the corelation and expected sign :

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

data corr\_base;

set \_null\_;

run;

%macro correlation\_stack();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

proc corr data= EM\_LIB.EM\_Qual\_MVI\_FINAL pearson outp=&variable.\_corel1;

var &variable. log\_PD;

/\*by country\_use;\*/

/\*ods output pearsonCorr = Corr;\*/

run;

data &variable.\_corel1;

length variable $50.;

set &variable.\_corel1;

variable="&variable.";

run;

data corr\_base;

set corr\_base &variable.\_corel1;

run;

%end;

%mend;

%correlation\_stack();

proc sql; create table corr\_table as select variable , \_TYPE\_,\_NAME\_,log\_PD from corr\_base

where \_TYPE\_='CORR' and \_NAME\_ ne 'log\_PD';

quit;

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

CODE AND MACRO TO CALCULATE ACCURACY RATIO FOR EACH QUALITATIVE FACTORS

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

data somers\_all;

Length variable $50.;

set \_null\_;

run;

options MPRINT MLOGIC SYMBOLGEN;

%macro somersd();

%do j=1 %to %sysfunc(countw(&final\_qual\_vars.));

%let variable=%scan(&final\_qual\_vars.,&j.);

ods trace on;

ods output association=somers;

proc logistic data = EM\_LIB.EM\_Qual\_MVI\_FINAL;

CLASS &variable./PARAM = REF;

model INTODEF = &variable.;

run;

ods trace off;

data temp1(rename=(label1=label nvalue1=nvalue));

set somers(keep= label1 nvalue1 );

run;

data temp2(rename=(label2=label nvalue2=nvalue));

set somers(keep= label2 nvalue2 );

run;

data temp;

set temp1 temp2;

run;

proc transpose data=temp out=transposed(drop=\_name\_);

id label;

run;

data transposed;

set transposed;

variable="&variable.";

run;

data somers\_all;

set somers\_all transposed;

run;

%end;

%mend;

%somersd();