Name- Binoyananda Nandi

LIBNAME FB "C:\SAS";

**RUN**;

**DATA** D V;

SET FB.BITS;

IF RANUNI(**789776**)<=**0.8** THEN OUTPUT D;

ELSE OUTPUT V;

**RUN**;

**PROC** **FREQ** DATA=D;

TABLES ACTIVE\_CUST;

**RUN**;

**%MACRO** FINE\_CLASS (DSN,DEPVAR,TOT\_RESP,TOT\_NRESP,CLASS\_VAR,GRPS);

PROC RANK DATA =&DSN. GROUPS=&GRPS. OUT=BIN\_DSN;

VAR &CLASS\_VAR.;

RANKS BIN;

RUN;

PROC SQL;

CREATE TABLE FINE\_CLASSED AS SELECT

COUNT(\*) AS NOBS,

MIN(&CLASS\_VAR.) AS MIN\_VAL,

MAX(&CLASS\_VAR.) AS MAX\_VAL,

SUM(CASE WHEN &DEPVAR. = **1** THEN **1** ELSE **0** END) AS RESP,

SUM(CASE WHEN &DEPVAR. = **0** THEN **1** ELSE **0** END) AS NRESP

FROM BIN\_DSN

GROUP BY BIN;

QUIT;

DATA FINE\_CLASSED1;

SET FINE\_CLASSED;

TOT\_RESP=&TOT\_RESP.;

TOT\_NRESP=&TOT\_NRESP.;

WOE=LOG(((RESP/TOT\_RESP)+**0.000000001**)/((NRESP/TOT\_nRESP)+**0.000000001**));

IV=((RESP/TOT\_RESP)-(NRESP/TOT\_NRESP))\*WOE\***100**;

RUN;

PROC PRINT DATA = FINE\_CLASSED1;RUN;

/\*PROC DELETE DATA = FINE\_CLASSED1;\*/

/\*PROC DELETE DATA = FINE\_CLASSED;\*/

/\*PROC DELETE DATA = BIN\_DSN;RUN;\*/

**%MEND**;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,INCOME,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,months\_on\_network,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,Num\_complaints,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,number\_plan\_changes,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,relocated,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,monthly\_bill,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,technical\_issues\_per\_month,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,Speed\_test\_result,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,INCOME\_WOE,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,MONTHS\_ON\_NETWORK\_WOE,**10**);

ODS HTML CLOSE;

ODS HTML FILE="C:\SAS\O.XLS";

%***FINE\_CLASS***(D,ACTIVE\_CUST,**46247**,**33715**,technical\_issues\_per\_month\_WOE,**10**);

ODS HTML CLOSE;

**DATA** FB.D;

SET FB.D;

IF INCOME>=**1135** AND INCOME<=**1605**THEN INCOME\_WOE=**0.519039**;

IF INCOME>=**1606** AND INCOME<=**1652**THEN INCOME\_WOE=**0.546851**;

IF INCOME>=**1653** AND INCOME<=**1694**THEN INCOME\_WOE=**0.137718**;

IF INCOME>=**1695** AND INCOME<=**1748**THEN INCOME\_WOE=**0.053099**;

IF INCOME>=**1749** AND INCOME<=**1805**THEN INCOME\_WOE=-**0.02134**;

IF INCOME>=**1806** AND INCOME<=**1846**THEN INCOME\_WOE=-**0.22379**;

IF INCOME>=**1847** AND INCOME<=**1883**THEN INCOME\_WOE=-**0.48575**;

IF INCOME>=**1884** AND INCOME<=**1944**THEN INCOME\_WOE=-**0.34644**;

IF INCOME>=**1945** AND INCOME<=**229821**THEN INCOME\_WOE=**0.006288**;

IF months\_on\_network>=-**20** AND months\_on\_network<=**37**THEN months\_on\_network\_WOE=-**0.1143**;

IF months\_on\_network>=**38** AND months\_on\_network<=**44**THEN months\_on\_network\_WOE=**0.133014**;

IF months\_on\_network>=**45** AND months\_on\_network<=**48**THEN months\_on\_network\_WOE=-**0.33112**;

IF months\_on\_network>=**49** AND months\_on\_network<=**51**THEN months\_on\_network\_WOE=-**0.50355**;

IF months\_on\_network>=**52** AND months\_on\_network<=**55**THEN months\_on\_network\_WOE=-**0.23197**;

IF months\_on\_network>=**56** AND months\_on\_network<=**61**THEN months\_on\_network\_WOE=-**0.01397**;

IF months\_on\_network>=**62** AND months\_on\_network<=**67**THEN months\_on\_network\_WOE=**0.046777**;

IF months\_on\_network>=**68** AND months\_on\_network<=**73**THEN months\_on\_network\_WOE=**0.146071**;

IF months\_on\_network>=**74** AND months\_on\_network<=**449**THEN months\_on\_network\_WOE=**0.53601**;

IF monthly\_bill >= **30** AND monthly\_bill <=**36** THEN monthly\_bill\_WOE = **0.075555**;

IF monthly\_bill >= **37** AND monthly\_bill <=**42** THEN monthly\_bill\_WOE = **0.076033**;

IF monthly\_bill >= **43** AND monthly\_bill <=**48** THEN monthly\_bill\_WOE = **0.076711**;

IF monthly\_bill >= **49** AND monthly\_bill <=**77** THEN monthly\_bill\_WOE = **0.036502**;

IF monthly\_bill >= **78** AND monthly\_bill <=**87** THEN monthly\_bill\_WOE = -**0.0228**;

IF monthly\_bill >= **88** AND monthly\_bill <=**118** THEN monthly\_bill\_WOE = -**0.02766**;

IF monthly\_bill >= **119** AND monthly\_bill <=**128** THEN monthly\_bill\_WOE = -**0.03228**;

IF monthly\_bill >= **129** AND monthly\_bill <=**160** THEN monthly\_bill\_WOE = -**0.06268**;

IF monthly\_bill >= **161** AND monthly\_bill <=**181** THEN monthly\_bill\_WOE = -**0.05935**;

IF Num\_complaints = **3** THEN Num\_complaints\_WOE = **0.48594**;

IF Num\_complaints = **4** THEN Num\_complaints\_WOE = **0.01871**;

IF Num\_complaints >= **5** AND Num\_complaints <=**6** THEN Num\_complaints\_WOE = -**0.70455**;

IF technical\_issues\_per\_month=**1** THEN technical\_issues\_per\_month\_WOE=**1.19309**;

IF technical\_issues\_per\_month>=**2** AND technical\_issues\_per\_month<=**3** THEN technical\_issues\_per\_month\_WOE=**0.73693**;

IF technical\_issues\_per\_month=**4** THEN technical\_issues\_per\_month\_WOE=-**0.32316**;

IF technical\_issues\_per\_month=**7** THEN technical\_issues\_per\_month\_WOE=**1.24645**;

**RUN**;

**PROC** **REG** DATA=FB.D;

MODEL active\_cust=

/\*INCOME\_WOE\*/

months\_on\_network\_WOE

technical\_issues\_per\_month\_WOE/

VIF COLLIN;

**QUIT**;

**RUN**;

ODS HTML FILE="C:\SAS\Q.xls";

**PROC** **LOGISTIC** DATA=FB.D descending;

MODEL active\_cust=

INCOME\_WOE

/\*months\_on\_network\_WOE\*/

technical\_issues\_per\_month\_WOE/ LACKFIT RSQ;

OUTPUT OUT = log.OUT P = PRED;

**QUIT**;

**RUN**;

ODS HTML CLOSE;

/\*

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

& &

& KS MACRO : SANKHA MUKHERJEE &

& &

&>>>>>>>>>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<<<<<<&

&>>>>>>>>>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<<<<<<&

& INSTRUCTIONS: &

& &

& DO NOT CHANGE ANYTHING IN THE MACRO &

& INPUT ONLY THE MACRO PARAMETERS AS &

& PER THE DEFINITIONS GIVEN BELOW &

& &

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

&>>>>>>>>>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<<<<<<&

& DEFINITIONS:MACRO PARAMETERS &

& &

& SET : name of data set which contains the score &

& SCORE : probability(score) variable name &

& RESP : name of the response variable &

& GROUP : number of equal groups in which you want &

& to break the data set e.g : deciles &

& TITLE : any name that you want to give to the &

& data set &

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

\*/

/\*rsubmit;\*/

**%macro** ksgcf(set,score,resp,group,title);

/\*\*\*\*\*\*\*Defining responders & nonresponders and creating Deciles\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

data t1;

set &set(keep=&score &resp);

nonresp=**1**-&resp;

run;

proc sort data = t1 ;by descending &score;run;

data \_null\_;

set t1;

call symput('size',\_n\_);

run;

data t1;

set t1;

decile=floor(**1**+&group\*(\_n\_-**1**)/&size);

run;

/\*\*\*\*\*\*\*\*Finding no of responders, nonresponders, minscore, maxscore and meanscore in each decile\*\*\*\*\*\*\*\*/

proc summary data = t1 nway missing;

var &score;

class decile;

output out=t2(drop=\_TYPE\_ \_FREQ\_) sum(nonresp &resp )=nonresp resp min(&score)=minscore max(&score)=maxscore

mean(&score)=meanscore;

run;

/\*\*\*\*\*\*\*\*Finding minscore, maxscore and meanscore for overall data set\*\*\*\*\*\*\*\*/

proc summary data = t1 nway missing;

var &score;

output out=t11(drop=\_TYPE\_ \_FREQ\_) min(&score)=minscore max(&score)=maxscore

mean(&score)=meanscore;

run;

data t2;

set t2 end=last;

/\*\*\*Formatting minscore and maxscore\*\*\*\*\*\*\*\*\*\*/

minscore=minscore\***1000**;format minscore **8.0**;

maxscore=maxscore\***1000**;format maxscore **8.0**;

/\*\*\*\*Calculating predicted response rate and actual response rate\*\*\*\*\*\*/

predrespr=meanscore\***100**;format predrespr **8.1**;

actrespr=(resp/(resp+nonresp))\***100**;format actrespr **8.1**;

/\*\*\*\*\*\*Calculating overall odds\*\*\*\*\*\*\*\*\*\*\*\*/

overall\_odds =nonresp/resp;

/\*\*\*\*\*\*\*\*Defining macro vars to pick up total numbers\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*calculating cumulative no of nonresponders\*\*\*\*\*\*\*/

cumnonresp+nonresp;

/\*\*\*\*\*\*\*\*calculating cumulative no of responders\*\*\*\*\*\*\*\*\*\*/

cumresp+resp;

/\*\*\*\*\*\*calculating cumulative no of observations\*\*\*\*\*\*\*\*\*\*/

cumtotal=cumnonresp+cumresp;

/\*\*\*\*\*\*\*\*\*calculating cumulative response rate\*\*\*\*\*\*\*/

cumrespr=(cumresp/cumtotal)\***100**;format cumrespr **8.1**;

/\*\*\*\*\*\*calculating cumulative odds\*\*\*\*\*\*\*\*\*/

cumodds=cumnonresp/cumresp;format cumodds **8.2**;

/\*\*\*\*\*\*defining macro vars for storing sum total values\*\*\*\*\*/

if last then do;

call symput('sumodds',cumodds);

call symput('sumnonresp',cumnonresp);

call symput('sumresp',cumresp);

call symput('sumcumresprate',cumrespr);

end;

run;

data t3(drop=cumresp cumnonresp cumodds y cumgini);

set t2 end=final;

/\*\*\*\*\*calculating information odds\*\*\*\*\*\*\*\*\*\*/

info\_odds=overall\_odds/&sumodds;format info\_odds **8.1**;

/\*\*\*\*\*\*\*formatting overall odds\*\*\*\*\*\*\*\*\*\*\*/

format overall\_odds **8.1**;

/\*\*\*\*calculating log odds\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

log\_odds=log(info\_odds);format log\_odds **8.1**;

/\*\*\*\*\*\*\*calculating probability of non response in each decile\*\*\*\*\*\*\*\*/

prob\_nonresp=nonresp/(nonresp+resp);format prob\_nonresp **8.2**;

/\*\*\*\*\*\*\*calculating chi square\*\*\*\*\*\*\*\*\*\*\*\*\*/

chi\_sq=((nonresp-(&sumnonresp/(&sumnonresp+&sumresp))\*(nonresp+resp))\*\***2**)/((&sumnonresp/(&sumnonresp+&sumresp))

\*(nonresp+resp))+((resp-(&sumresp/(&sumnonresp+&sumresp))\*(nonresp+resp))\*\***2**)/((&sumresp/(&sumnonresp+&sumresp))

\*(nonresp+resp));format chi\_sq **8.1**;

/\*\*\*\*\*\*\*\*calculating cumulative chi square\*\*\*\*\*\*\*\*\*\*/

cumchi\_sq+chi\_sq;

/\*\*\*\*\*\*\*calculating percentage of nonresponders in each decile\*\*\*\*\*\*\*\*\*/

pernonresp=(nonresp/&sumnonresp)\***100**;format pernonresp **8.1**;

/\*\*\*\*\*\*\*calculating percentage of responders in each decile\*\*\*\*\*\*\*\*\*/

perresp=(resp/&sumresp)\***100**;format perresp **8.1**;

/\*\*\*\*\*\*\*calculating cumulative percentage of nonresponders in each decile\*\*\*\*\*\*\*\*\*/

cumpernonresp+pernonresp;format cumpernonresp **8.1**;

/\*\*\*\*\*\*\*calculating cumulative percentage of responders in each decile\*\*\*\*\*\*\*\*\*/

cumperresp+perresp;format cumperresp **8.1**;

/\*\*\*\*\*\*\*\*\*calculating percentage of observations in each decile\*\*\*\*\*\*\*\*\*\*\*\*/

perobs=((nonresp+resp)/(&sumnonresp+&sumresp))\***100**;format perobs **8.0**;

/\*\*\*\*\*\*\*\*\*calculating cumulative percentage of observations in each decile\*\*\*\*\*\*\*\*\*\*\*\*/

cumperobs+perobs;format cumperobs **8.0**;

/\*\*\*\*\*\*\*\*\*\*calculating individual lift obtained from each decile\*\*\*\*\*\*\*\*\*/

lift=(actrespr/&sumcumresprate)\***100**;format lift **8.0**;

/\*\*\*\*\*\*\*\*calculating cumulative lift\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

cumlift=(cumrespr/&sumcumresprate)\***100**;format cumlift **8.0**;

/\*\*\*\*\*\*\*\*\*\*\*calculating KS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ks=abs(cumperresp-cumpernonresp);format ks **8.1**;

lg1=lag(resp);

if \_n\_=**1** then flag=**1**;else flag=(resp<=lg1);

if flag=**0** then break=\_n\_;

y=lag(cumperresp);

gini=((sum(cumperresp,y)/**2**)\*pernonresp)/(**100**\*\***2**);format gini **8.2**;

cumgini+gini;

if final then do;

call symput('sumgini',cumgini);

end;

predresp=meanscore\*(resp+nonresp);

diff=resp-predresp;

gofcell=(diff\*\***2**)/((resp+nonresp)\*meanscore\*(**1**-meanscore));

gof+gofcell;format gof **8.1**;

meanscore=meanscore\***1000**;format meanscore **8.0**;

run;

data t3;

set t3;

totalgini=&sumgini-**0.5**;format totalgini **8.4**;

run;

proc transpose data =t3 out=t4;var break;run;

data t4(keep=sat\_rank ranking);

set t4(drop=\_NAME\_);

array a{&group} col1 - col&group;

do i=**1** to &group ;

if a{i}>**0** then leave;

end;

if i=(&group+**1**) then sat\_rank='all';else sat\_rank=i-**1**;

if sat\_rank='all' then ranking='SATISFACTORY ';else ranking='NOT SATISFACTORY';

run;

proc transpose data = t3 out=t5;var ks ;run;

data t5(keep=ks maxksdec);

set t5(drop=\_NAME\_);

array ks1{&group} col1 - col&group;

format ks **8.1**;

ks=**0**;

do i=**1** to &group;

if ks1{i}> ks then ks=ks1{i};else leave;

end;

maxksdec=i-**1**;

run;

proc transpose data = t3 out=t6;var totalgini;run;

data t6;

set t6(keep=col1);

rename col1=gini;

run;

proc transpose data = t3 out=t7;var gof ;run;

data t7;

set t7(keep=col&group);

rename col&group=gof;

run;

data mrg;

merge t4 t5 t6 t7;

run;

data total(drop=cumrespr);

set t2(obs=**1**);

nonresp=&sumnonresp;

resp=&sumresp;

actrespr=(resp/(resp+nonresp))\***100**;format actrespr **8.1**;

overall\_odds =nonresp/resp;

info\_odds=overall\_odds/&sumodds;

format overall\_odds **8.1**;

format info\_odds **8.1**;

log\_odds=log(info\_odds);format log\_odds **8.1**;

prob\_nonresp=nonresp/(nonresp+resp);format prob\_nonresp **8.2**;

pernonresp=(nonresp/nonresp)\***100**;format pernonresp **8.1**;

perresp=(resp/resp)\***100**;format perresp **8.1**;

perobs=((nonresp+resp)/(&sumnonresp+&sumresp))\***100**;format perobs **8.0**;

run;

data total1;

set t11;

minscore=minscore\***1000**;format minscore **8.0**;

maxscore=maxscore\***1000**;format maxscore **8.0**;

predrespr=meanscore\***100**;format predrespr **8.1**;

meanscore=meanscore\***1000**;format meanscore **8.0**;

run;

data total11(keep=cumchi\_sq cumgini);

set t3;

cumchisq+chi\_sq;

cumgini+gini;

if \_n\_=&group;

run;

data total111;

set t5(keep=ks);

run;

data total1111;

set t7(keep=gof);

run;

data total;

merge total total1 total11 total111 total1111;

rename cumchi\_sq=chi\_sq;

rename cumgini=gini;

run;

data t3(drop=decile);

set t3 total ;

if \_n\_=&group+**1** then decile=**9999**;

decile1=put(decile,**8.**);format decile1 $8.;

run;

data t3;

set t3;

rename decile1=decile;

decile1=left(trim(decile1));

if decile1="9999" then decile1="Total";

run;

proc print data = t3 noobs;

title "&title";

var decile nonresp resp actrespr predrespr minscore maxscore meanscore overall\_odds info\_odds log\_odds

prob\_nonresp chi\_sq pernonresp perresp cumpernonresp cumperresp perobs cumperobs cumrespr lift cumlift gini ks gof;

run;

proc print data = mrg noobs;

var ranking sat\_rank ks maxksdec gini gof;

run;

**%mend**;

ODS HTML FILE="C:\SAS\M.xls";

%***ksgcf***(log.out,pred,active\_cust,**5**,RO);

ods html close;

**DATA** log.OUT;

SET log.OUT;

SCORE = ROUND(PRED\***1000**);

**RUN**;

ODS HTML FILE = "C:\SAS\N.xls";

**PROC** **TTEST** DATA = log.OUT;

VAR SCORE;

CLASS active\_cust ;

**RUN**;

ODS HTML CLOSE;

ODS HTML FILE = "C:\SAS.xls";

**PROC** **LOGISTIC** DATA = log.VAL DESCENDING ;

MODEL ACTIVE\_CUST=

INCOME\_WOE

/\*months\_on\_network\_WOE\*/

technical\_issues\_per\_month\_WOE/ LACKFIT RSQ;

OUTPUT OUT = log.OUT2 P = PRED1;

**RUN**;

**QUIT**;

ODS HTML CLOSE;

ODS HTML FILE = "C:\SAS\V.xls";

%***KSGCF***(log.OUT2,PRED1,ACTIVE\_CUST,**5**,RO);

**run**;

ODS HTML CLOSE;

**DATA** log.val;

SET log.val;

L=**0.2486**+**1.2643**\*INCOME\_WOE+**0.9988**\*technical\_issues\_per\_month\_WOE;

P = EXP(L)/(**1**+EXP(L));

**RUN**;

ODS HTML FILE = "C:\SAS\FBOUTPUT.XLS";

%***KSGCF***(Log.val,P,ACTIVE\_CUST,**10**,RO);

ODS HTML CLOSE;