ASSIGNMENT

ASS06

DATA PREPARATION – STA6714

DONE BY:

SUSHMITHA MANI

**QUESTION 1**

*Use the data to produce a ROC Curve based on decile.*

**CODE:**

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.1 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.1 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.1 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.1 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.2 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.2 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.2 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.2 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.3 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.3 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.3 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.3 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.4 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.4 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.4 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.4 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.5 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.5 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.5 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.5 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.6 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.6 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.6 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.6 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.7 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.7 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.7 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.7 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.8 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.8 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.8 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.8 and True\_Target=0;

quit;

title 'ROC curve based on deciles';

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.9 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.9 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.9 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.9 and True\_Target=0;

quit;

/\*creating and inserting in decile table\*/

proc sql;

create table decile (Decile\_Level numeric, Cum\_TPC numeric, Cum\_FPC numeric,

Cum\_FNC numeric, Cum\_TNC numeric);

insert into decile values (1, 49929, 49816, 16, 239);

insert into decile values (2, 49345, 47386, 600, 2669);

insert into decile values (3, 47071, 40677, 2874, 9378);

insert into decile values (4, 41958, 30607, 7987, 19448);

insert into decile values (5, 31951, 18725, 17994, 31330);

insert into decile values (6, 17816, 7201, 32129, 42854);

insert into decile values (7, 9737, 2263, 40208, 47792);

insert into decile values (8, 6565, 1040, 43380, 49015);

insert into decile values (9, 670, 57, 49275, 49998);

select \* from decile;

quit;

/\*additing sensitivity and 1-specificity columns\*/

proc sql;

alter table decile add Sensitivity numeric, False\_Alarm\_Rate numeric;

quit;

data decilesesp;

set decile;

Sensitivity = Cum\_TPC/(Cum\_TPC + Cum\_FNC);

False\_Alarm\_Rate = 1-(Cum\_TNC/(Cum\_TNC + Cum\_FPC));

run;

proc print data=decilesesp;

run;

proc logistic data=work.import noprint;

model True\_Target(event='1') = Posterior\_Probability;

output out=LogiOut predicted=PredProb;

run;

proc rank data=LogiOut out=LogiDecile groups=10;

var PredProb;

ranks Decile;

run;

proc means data=LogiDecile noprint;

class Decile;

types Decile;

var True\_Target PredProb;

output out=LogiDecileOut mean=True\_TargetMean PredProbMean

lclm=True\_TargetLower uclm=True\_TargetUpper;

run;

proc logistic data=LogiDecile;

model True\_Target(event='1')=PredProb/nofit;

roc 'Predictions' pred=PredProb;

ods select ROCcurve;

run;

proc logistic data=work.import noprint;

model True\_Target(event='0') = Posterior\_Probability;

output out=LogiOut predicted=PredProb;

run;

proc rank data=LogiOut out=LogiDecile0 groups=10;

var PredProb;

ranks Decile;

run;

proc means data=LogiDecile0 noprint;

class Decile;

types Decile;

var True\_Target PredProb;

output out=LogiDecileOut mean=True\_TargetMean PredProbMean

lclm=True\_TargetLower uclm=True\_TargetUpper;

run;

proc logistic data=LogiDecile0;

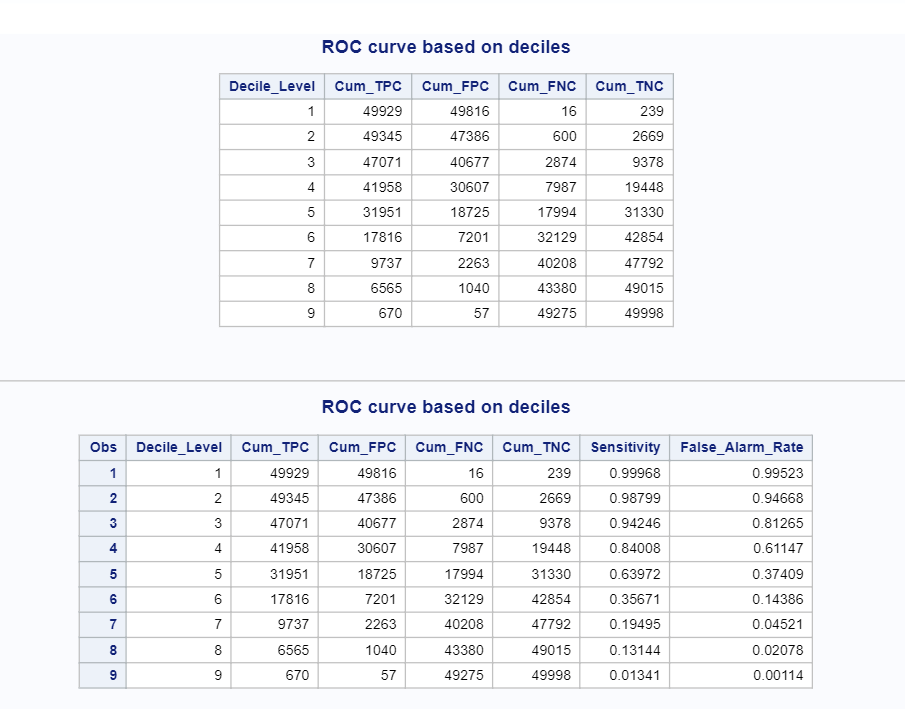
model True\_Target(event='0')=PredProb/nofit;

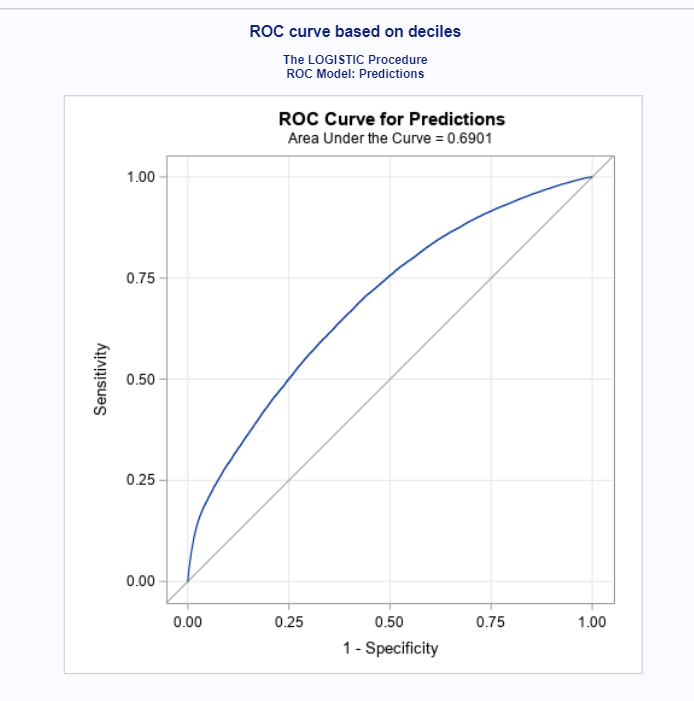
roc 'Predictions' pred=PredProb;

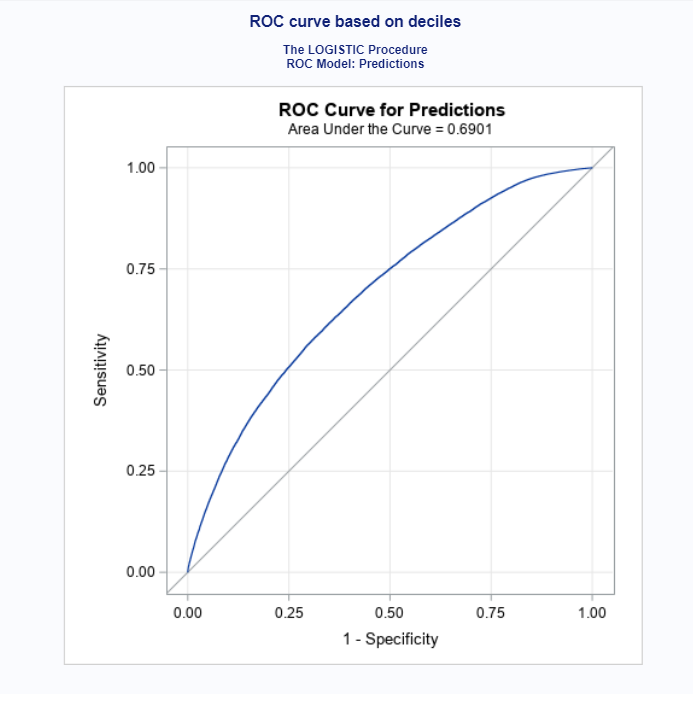
ods select ROCcurve;

run;

**OUTPUT:**

****

****

****

**QUESTION 2:**

*Use the same data to produce a lift chart at percentile level.*

**CODE:**

procrankdata=work.assign\_dataout=percentiles ties=low

descendinggroups=100;

varPosterior\_Probability;

ranks percentile;

run;

procsql;

select sum(True\_Target) into: total\_hits

fromwork.assign\_data;

createtable lift as

select

sum(True\_Target)/&total\_hitsastrue\_positive\_rate

,percentile + 1as percentile

from percentiles

groupby percentile

orderby percentile

;

quit;

datacum\_lift;

set lift;

cum\_positive\_rate + true\_positive\_rate;

cum\_lift=cum\_positive\_rate/(percentile/100);

run;

procgplotdata=cum\_lift;

title'Cummulative Lift Chart';

symboli=spline;

plotcum\_lift\*percentile /grid;

run;

quit;

procsgplotdata=cum\_lift;

scatterx=percentile y=cum\_lift;

run;

procgplotdata=cum\_lift;

title ‘Cumulative Lift Chart’;

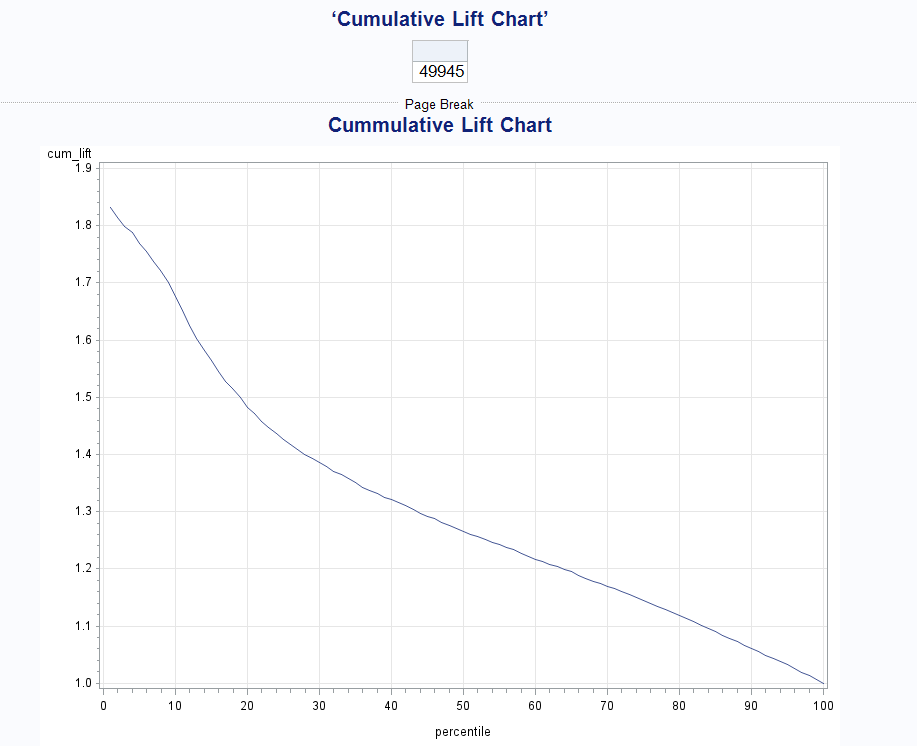
symboli=spline;

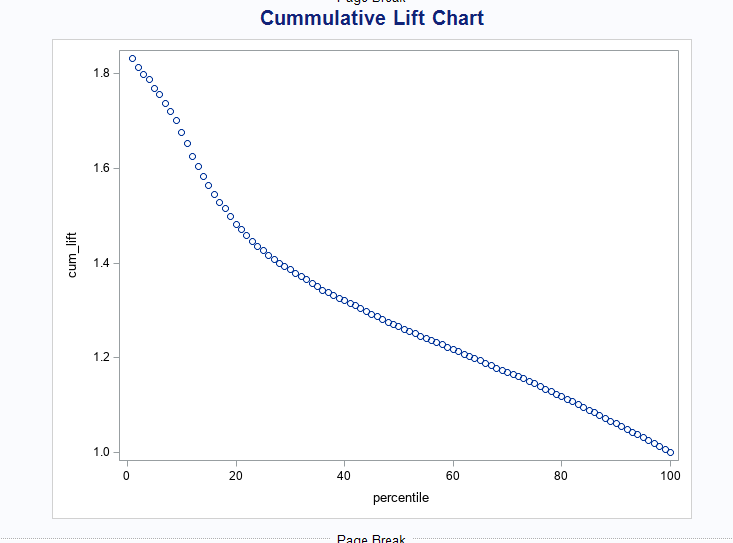
plotcum\_lift\*percentile /grid;

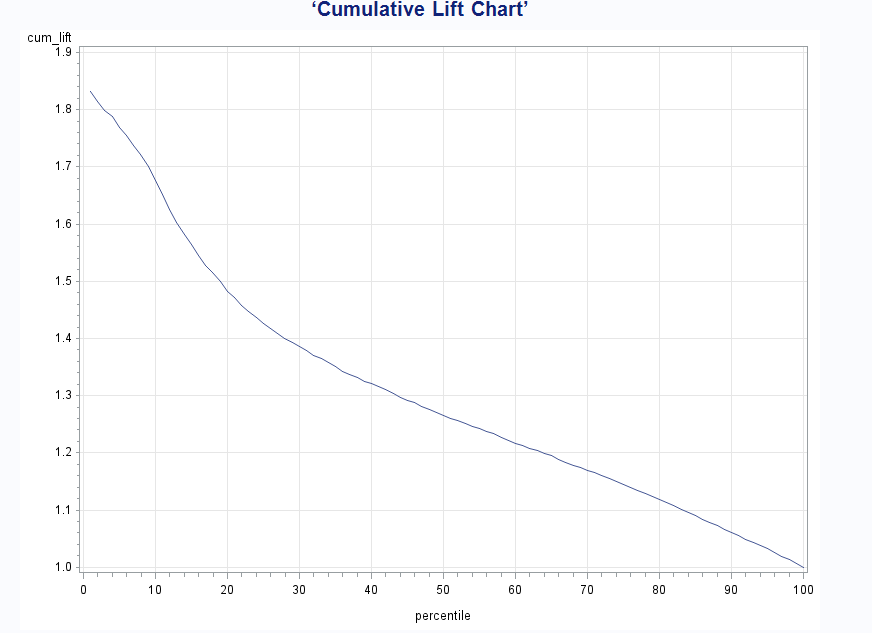
run;

quit;

**OUTPUT:**







**QUESTION 3:**

*Produce one confusion matrix for each of the following cut-off probability 0.1, 0.2, and 0.3 using the same data.*

**CODE:**

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.1 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.1 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.1 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.1 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.2 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.2 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.2 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.2 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.3 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability>0.3 and True\_Target=0;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.3 and True\_Target=1;

quit;

proc sql noprint;

select count(\*) from work.import where Posterior\_Probability<0.3 and True\_Target=0;

quit;

proc sql;

create table Matrix (Decile\_Level numeric, Cum\_TPC numeric, Cum\_FPC numeric,

Cum\_FNC numeric, Cum\_TNC numeric);

insert into matrix values (1, 49929, 16, 239, 49816);

insert into matrix values (2, 49345, 600, 2669, 47386);

insert into matrix values (3, 47071, 2874, 9378, 40677);

select \* from matrix;

quit;

proc sql;

title 'Confusion Matrix with cut-off Probability 0.1';

create table confusion1 (TrueState char(10), Positive num, Negative num);

insert into confusion1 values('True', 49929, 16);

insert into confusion1 values('False', 49816, 239);

select\*

from confusion1;

quit;

proc sql;

title 'Confusion Matrix with cut-off Probability 0.2';

create table confusion2 (TrueState char(10), Positive num, Negative num);

insert into confusion2 values('True', 49345, 600);

insert into confusion2 values('False',47386, 2669);

select\*

from confusion3;

quit;

proc sql;

title 'Confusion Matrix with cut-off Probability 0.3';

create table confusion3 (TrueState char(10), Positive num, Negative num);

insert into confusion3 values('True', 47071, 2874);

insert into confusion3 values('False', 40677, 9378);

select\*

from confusion3;

quit;

**OUTPUT:**

