

Supplementary material: SAS Code

```
/* Weighted logistic regression */
proc logistic data = TrainData;
    model &y (event='1') = &x1 &x2 &x3 &x4 &x5 /
    selection= stepwise; weight &w;
    score data = TrainData out = TrainDataScored;
    score data = ValidData out = ValidDataScored;
run;

/* Calculate Gini on Train */
proc freq data = TrainDataScored;
    table &y * p_1 / measures;
    weight &w;
    ods output measures=data;
run;

* to increase the number of decimal places;
data data;
    set data;
    format value 24.10 ;
    informat value 24.10;
run;

/* Calculate Gini on Valid */
proc freq data = ValidDataScored;
    table &y * p_1 / measures;
    weight &w;
    ods output measures=data;
run;

* to increase the number of decimal places;
data data;
    set data;
    format value 24.10 ;
    informat value 24.10;
run;

/* Calculate MSE on Train */
data test;
    set TrainDataScored;
    squarederror = (lgd_est-p_1)* (lgd_est-p_1);
run;
proc summary data = test nway;
    var squarederror;
    output out = mse sum =;
run;
data mse;
    set mse;
```

```

        mse = squarederror/_FREQ_;
run;
/* Calculate MSE on Valid */
data test;
set ValidDataScored;
        squarederror = (lgd_est-p_1)* (lgd_est-p_1);
run;
proc summary data = test nway;
        var squarederror;
        output out = mse sum = ;
run;
data mse;
        set mse;
        mse = squarederror/_FREQ_;
run;
/* GEE with independent correlation */
PROC GENMOD DATA=TrainData;
CLASS account_number ;
model &y (event='1') = &x1 &x2 &x3 &x4 &x5
/DIST=BIN LINK=LOGIT ;
REPEATED SUBJECT= account_number / TYPE=IND CORRW;
        weight &w;
        store out = GEE1;
run;

/* Calculate Gini on Train */
proc plm source=GEE1;
        score data=TrainData out=preds pred=pred
                lclm=lower uclm=upper/ ilink;
run;
proc freq data = preds;
        table &y * pred / measures ;
        weight &w;
        ods output measures=data;
run;
* to increase the number of decimal places;
data data;
        set data;
        format value 24.10 ;
        informat value 24.10;
run;
/* Calculate MSE on Train */
data test;
set preds;
        squarederror = (lgd_est-p_1)* (lgd_est-p_1);

```

```

run;
proc summary data = test nway;
    var squarederror;
    output out = MSE sum = ;
run;
data mse ;
    set mse;
    mse = squarederror/_FREQ_;
run;
/* Calculate Gini on Valid */
proc plm source=GEE1;
    score data=TrainData out=preds pred=pred
        lclm=lower uclm=upper/ ilink;
run;
proc freq data = preds;
    table &y * pred / measures ;
    weight &w;
    ods output measures=data;
run;
* to increase the number of decimal places;
data data;
    set data;
    format value 24.10 ;
    informat value 24.10;
run;
/* Calculate MSE on Valid */
data test;
    set preds;
    squarederror = (lgd_est-p_1)* (lgd_est-p_1);
run;
proc summary data = test nway;
    var squarederror;
    output out = MSE sum = ;
run;
data mse ;
    set mse;
    mse = squarederror/_FREQ_;
run;
/* GEE with AR1 correlation */
PROC GENMOD DATA=TrainData;
CLASS account_number ;
model &y (event='1') = &x1 &x2 &x3 &x4 &x5
/DIST=BIN LINK=LOGIT ;
REPEATED SUBJECT= account_number / TYPE= AR(1) CORRW;
weight &w;

```

```

        store out = GEE1;
run;
/* Calculate Gini on Train */
proc plm source=GEE1;
    score data=TrainData out=preds pred=pred
        lclm=lower uclm=upper/ ilink;
run;
proc freq data = preds;
    table &y * pred / measures ;
    weight &w;
    ods output measures=data;
run;
* to increase the number of decimal places;
data data;
    set data;
    format value 24.10 ;
    informat value 24.10;
run;
/* Calculate MSE on Train */
data test;
set preds;
    squarederror = (lgd_est-p_1)* (lgd_est-p_1);
run;
proc summary data = test nway;
    var squarederror;
    output out = MSE sum = ;
run;
data mse ;
    set mse;
    mse = squarederror/_FREQ_;
run;
/* Calculate Gini on Valid */
proc plm source=GEE1;
    score data=TrainData out=preds pred=pred
        lclm=lower uclm=upper/ ilink;
run;
proc freq data = preds;
    table &y * pred / measures ;
    weight &w;
    ods output measures=data;
run;
* to increase the number of decimal places;
data data;
    set data;
    format value 24.10 ;

```

```

        informat value 24.10;
run;
/* Calculate MSE on Valid */
data test;
set preds;
    squarederror = (lgd_est-p_1)* (lgd_est-p_1);
run;
proc summary data = test nway;
    var squarederror;
    output out = MSE sum = ;
run;
data mse ;
    set mse;
    mse = squarederror/_FREQ_;
run;

*Decision Tree 1: Default settings;
ods graphics on;
proc hpsplit data=TrainData leafsize=100;
weight &w;
model &y = 1&x1 &x2 &x3 &x4 &x5;;
code file='&path.\hpspl1.sas'; run;

data DecisionTreeValid1;
    set ValidData;
%include "&path.\hpspl1.sas";
run;
/* Calculate MSE on Valid */
data test1;
set DecisionTreeValid1;
    squarederror = (lgd_est - P_Binary_outcome)* (lgd_est - P_Binary_outcome);
run;
proc summary data = test1 nway;
    var squarederror;
    output out = mse sum = ;
run;
data mse1;
    set mse;
    mse = squarederror/_FREQ_;
run;

```

```

*Decision Tree 2: Prune on ASE;
ods graphics on;
proc hpsplit data=TrainData leafsize=100;
weight &w;
model &y = l&x1 &x2 &x3 &x4 &x5;
prune ase / N <= 10; ;
code file='&path.\hpspl2.sas'; run;

data DecisionTreeValid2;
    set ValidData;
%include "&path.\hpspl2.sas";
run;
/* Calculate MSE on Valid */
data test2;
set DecisionTreeValid2;
    squarederror = (lgd_est - P_Binary_outcome)* (lgd_est - P_Binary_outcome);
run;
proc summary data = test2 nway;
    var squarederror;
    output out = mse sum = ;
run;
data mse2;
    set mse;
    mse = squarederror/_FREQ_;
run;

*Decision Tree 3: no pruning;
ods graphics on;
proc hpsplit data=TrainData leafsize=100;
weight &w;
model &y = l&x1 &x2 &x3 &x4 &x5;
prune none ;
code file='&path.\hpspl3.sas'; run;

data DecisionTreeValid2;
    set ValidData;
%include "&path.\hpspl3.sas";
run;
/* Calculate MSE on Valid */
data test3;
set DecisionTreeValid3;
    squarederror = (lgd_est - P_Binary_outcome)* (lgd_est - P_Binary_outcome);
run;
proc summary data = test3 nway;

```

```
var squarederror;  
output out = mse sum = ;  
run;  
data mse3;  
set mse;  
mse = squarederror/_FREQ_;  
run;
```