

Lending a hand with SAS® Software - An Application in the Banking Industry

Deborah J. Blair, CoreStates Bank, Wilmington, DE
W. Lily Hadinoto, CoreStates Bank, Philadelphia, PA

ABSTRACT

Fair Lending issues are of paramount concern to the banking industry. This paper describes one technique that CoreStates Bank uses to monitor its' lending practices. In this approach, we use JCL, Job Control Language and Base SAS on an IBM mainframe. Geared towards the novice SAS programmer, two procedures, PROC SQL and PROC Report are key to this program. This paper details a real-world problem and offers a simple, yet innovative solution.

INTRODUCTION

The fair lending practices of all lending organizations: banks, mortgage companies, credit card issuers, are under constant examination by current and potential customers, by community groups as well as by regulatory agencies of the Federal Government. CoreStates Bank is all three, a bank, a mortgage company and a credit card issuer. CoreStates Bank has always been committed to providing the best, quality products to all of it's customers. They currently use industry-wide standards and methods for decisioning an application. After an application is received, it is given a credit score, which among other things, is based on the credit history of the applicant(s).

CoreStates' goal was to take a proactive role in assisting their lenders with the decision-making process, as well as to monitor it's own lending practices. CoreStates wanted a method and/or a technique that would help it's lenders evaluate an application and decision an application as non-judgementally as possible. CoreStates also strove for a solution that would identify any potential inconsistencies in lending practices.

The solution was a SAS program that matches and compares applications that were turned-down to applications that have been approved. The matching criteria is based on specific characteristics found in both applications.

In the past, CoreStates has used Base SAS mostly in a test environment for ad hoc reporting. This program is unique in that it is set up to run in our production environment, on a nightly basis. The report is automatically generated and sent to a remote mainframe printer Monday through Friday of each work week. The lenders have access to the report at the beginning of each day. This was crucial to the success of the project in that it allowed for quick access to the data on our Loan Application System, ACAPs.

THE PROGRAM

The JCL

The JCL is very basic. It accesses data using DD statements. The data represents flat files which are extracts from ACAPs. The output can be sent directly to a remoter printer by using the

Destination or DEST parameter in the EXEC SAS statement. Using the Affinity sub-parameter within the Unit parameter, or UNIT=AFF=DDname, and specifying the previous DDName, you can use the same tape drive successively. In this program, only one tape drive is requested, instead of two. This can be crucial to efficiency in some environments.

```
//CL9741BY JOB (009978),'HADINOTO-L.1-9-2-4',
//   MSGLEVEL=(2,1),NOTIFY=CL9741,
//   MSGCLASS=N,CLASS=I
//SASJOB EXEC SAS,COPIES=1,TIME=180,
//   SPRT=V,REGION=5M
//*
//INDD1 DD DSN=PNB1.LAP.COMBINE,DISP=SHR
//INDD2 DD DSN=PNB1.LAP.BEFORE.QAFAPV26,
//   DISP=SHR,UNIT=AFF=INDD1
//*
//WORK DD UNIT=SYSDA,SPACE=(CYL,(200,100))
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
/*
```

Macro Variables

Before we run the prgram we must determine if the prior work day was a holiday, and also if it was a Saturday, Sunday or Monday.

```
OPTIONS MLOGIC MACROGEN SYMBOLGEN;
```

```
*-----*;
  WAS YESTERDAY A MONDAY AND A HOLIDAY? IF
  YES, ENTER A YES FOR THE MON_HOLD VARIABLE.
  IF NOT, LEAVE IT AS A NO.
*-----*;
%LET MON_HOLD = NO;

*-----*;
  WAS YESTERDAY A HOLIDAY BUT NOT A SATURDAY,
  SUNDAY OR MONDAY? IF YES, ENTER A YES TO THE
  VARIABLE HOLIDAY BELOW. IF NOT, LEAVE IT AS A NO.
*-----*;
%LET HOLIDAY = NO;

*-----*;
  SET THE START AND LAST DATE.
  IF YESTERDAY WAS MONDAY AND A HOLIDAY OR
  SUNDAY, THEN THE START DATE WOULD BE FRIDAY'S
  DATE AND LAST DATE WOULD BE SATURDAY'S DATE.
  IF YESTERDAY WAS A HOLIDAY BUT NOT SATURDAY,
  SUNDAY OR MONDAY, THEN THE START AND LAST DATE
  WOULD BE THE DAY BEFORE THE HOLIDAY.
  OTHERWISE, THE START AND LAST DATES WOULD BE
  THE SAME, YESTERDAY'S DATE.
*-----*;
```

```

%MACRO DAYS;
  %IF &MON_HOLD = YES %THEN %DO;
STARTDTE = TODAY() - 4;
LASTDATE = TODAY() - 3;
%END;

  %ELSE %IF HOLIDAY = YES %THEN %DO;
  STARTDTE = TODAY() - 2;
  LASTDATE = STARTDTE;
%END;

  %ELSE %IF %UPCASE(&SYSDAY) = MONDAY
  %THEN %DO;
  STARTDTE = TODAY() - 3;
  LASTDATE = TODAY() - 2;
%END;

  %ELSE %DO;
  STARTDTE = TODAY() - 1;
  LASTDATE = STARTDTE;
%END;
%MEND DAYS;

%MACRO TITLES;
TITLE1 *** CONFIDENTIAL: FOR INTERNAL USE
ONLY ***;
TITLE3 'COMPARISON REPORTING FOR ALL BANKS';
TITLE4 'BY PRODUCT, CELL, AND DECISION CODE';
TITLE5 'APPLICATIONS WERE RECEIVED: &FDATE';
FOOTNOTE1 'PREPARED BY: RETAIL CREDIT RISK
TECHNOLOGY(DJB,WLH) ' ;
FOOTNOTE2 'PROGRAM NAME:
PNB4.CL9741.FAIR(NEWMATC4)';
FOOTNOTE4 *** CONFIDENTIAL: FOR INTERNAL USE
ONLY ***;
%MEND TITLES;

Data Steps and Procedures

-----*;
  READ IN PNB1.LAP.COMBINE
-----*;

DATA LAPS1;
  DROP DECISION ;
  INFILE INDD1 MISSEVER;
  INPUT @275 LOCATION $6.
         @883 APRVPROD $5. @;

  IF SUBSTR(LOCATION,1,2) = '01' AND
  SUBSTR(LOCATION,3,2) IN ('01','02','03','04');

-----*;
  INCLUDE ONLY THE FOLLOWING PRODUCTS:
  HOME EQUITY LOANS AND LINES, UNSECURED
  PERSONAL LOANS APPLICATIONS
-----*;
IF APRVPROD IN ('LHE1', 'LSL1', 'LPR1', 'LPR2', 'LUM1');

  INPUT @1 APPID $15.
         @402 DATE_ENT PD5.
         @431 DECISION PD5.
         @440 ADJ_CODE $1.

```

```

@441 ADJ_USER $8.
@888 APP_AMT PD6.2
@1680 TOT_INC PD7.2
@1753 APP_DI PD4.3
@2068 BSCORE PD3.
@2493 GENDER $1.
@2494 RACE $1.
@3343 JUD_TD_1 $3.
@3346 JUD_TD_2 $3.
@3414 PROCEED $1.
@3490 EST_LTV PD3.3
@3493 ACT_LTV PD3.3
@3601 OR_USER $8.
@3609 OR_REASN $3.
;

-----*;
CREATE SAS DATES
-----*;
DATE_REC = INPUT(PUT(DATE_ENT,10.),YYMMDD10.);
ADJ_DATE = INPUT(PUT(DECISION,10.),YYMMDD10.);

-----*;
CALL "DAYS" MACRO DEFINED ABOVE TO DETERMINE
THE APPLICATION DATE TO BE REPORTED.
-----*;
%DAYS

-----*;
RUN SAS MACRO FDATE (FORMAT DATE) TO PRINT
THE LAST DATE OF APPLICATIONS RECEIVED, ON THE
TITLE
-----*;
CALL SYMPUT('FDATE',PUT(LASTDATE,WORDDATE.));

-----*;
COMPARE START AND LAST DATE OF DECLINED
APPLICATIONS TO APPLICATIONS APPROVED WITHIN
THE LAST 30 DAYS AND CREATE THE DATE RANGE USED
IN THE COMPARISON.
-----*;
RANGE = STARTDTE - 30;
IF RANGE <= DATE_REC <= LASTDATE;

-----*;
DELETE APPLICATIONS THAT HAVE NOT BEEN
DECISIONED, TRAINING OR RELATED APPLICATIONS
-----*;
IF (ADJ_CODE = ' ) OR
(SUBSTR(APPID,1,2) IN ('TR', 'MI')) OR
(SUBSTR(APPID,14,1) = 'R') THEN DELETE;

-----*;
CREATE A FORMAT CALLED $APPLID TO PULL
APPLICATIONS FROM THE SCORING RECORD
-----*;

```

¹SAS® Guide to Macro Processing, Version 6, Second Edition

```

PROC SORT DATA = LAPS1 NODUPKEY
  OUT = CNTLACCT;
  BY APPID;

DATA CNTLACCT (RENAME = (APPID = START));
  SET CNTLACCT (KEEP = APPID);

  FMTNAME = '$APPLID';
  LABEL = 'OK';
  TYPE = 'C';

PROC FORMAT CNTLIN = CNTLACCT;

*-----*;
READ IN SCORING RECORD
*-----*;
DATA CREDIT (KEEP = APPID CRD_NAME SCORE
              SYS_TD_1);
  INFILE INDD2 MISSEVER;
  INPUT @1 APPID $15.
        @21 RECTYPE $2. @;

  IF RECTYPE = '41' AND PUT(APPID,$APPLID.) = 'OK';

  INPUT @571 CRD_NAME $8.
        @588 SCORE PD3.
        @592 SYS_TD_1 $3.
        ;

PROC SORT DATA = LAPS1;
  BY APPID;

PROC SORT DATA=CREDIT;
  BY APPID;

DATA LAPS2 (DROP = LOCATION);
  MERGE LAPS1(IN = A) CREDIT;
  BY APPID;
  IF A = 1;
  IF CRD_NAME IN ('US1002D','SE1002D');

*-----*;
IF APPLICATION WAS AN AUTOMATIC DECLINE
(DECLINED BY SYSTEM)
THEN THE SYSTEM TURN-DOWN REASON SHOULD BE
LISTED UNDER THE TURN-DOWN CODE COLUMN.
*-----*;
  IF COMPRESS(ADJ_USER) EQ 'SYSTEM' THEN
    D_TDCODE = SYS_TD_1;

    ELSE D_TDCODE = JUD_TD_1;

*-----*;
CREATE YEARLY INCOME FROM GROSS MONTHLY
INCOME.
*-----*;
  YR_INC = ROUND(TOT_INC * 12);

```

```

*-----*;
CLASSIFY BUREAU SCORES INTO LOW, MEDIUM AND
HIGH RISK GROUPS
*-----*;
IF 10 <= BSCORE <= 649 THEN BURSCR = 'LOW';
  ELSE IF 650 <= BSCORE <= 679
    THEN BURSCR = 'MED';
  ELSE IF 680 <= BSCORE <= 1000
    THEN BURSCR = 'HIGH';
  ELSE BURSCR = 'NONE';

*-----*;
ASSIGN VALUES TO PASSREC (RECOMMENDATION) AND
CELL VARIABLES
*-----*;
IF CRD_NAME = 'SE1002D' THEN DO;

SELECT (BURSCR);
  WHEN ('NONE') DO;
    CELL = 'NO BSCORE';
    IF SCORE >= 250 THEN PASSREC = 'A';
    ELSE IF 215 <= SCORE <= 249 THEN PASSREC = 'T';
    ELSE PASSREC = 'R';
  END;

  WHEN ('LOW') DO;
    IF SCORE >= 250 THEN DO;
      PASSREC = 'A';
      CELL = '3HL';
    END;

    ELSE IF 215 <= SCORE <= 249 THEN DO;
      PASSREC = 'T';
      CELL = '6ML';
    END;

    ELSE DO;
      PASSREC = 'R';
      CELL = '9LL';
    END;
  END;

  WHEN ('MED') DO;
    IF SCORE >= 250 THEN DO;
      PASSREC = 'A';
      CELL = '2HM';
    END;

    ELSE IF 215 <= SCORE <= 249 THEN DO;
      PASSREC = 'A';
      CELL = '5MM';
    END;

    ELSE DO;
      PASSREC = 'R';
      CELL = '8LM';
    END;
  END;
END;

```

```

WHEN ('HIGH') DO;
  IF SCORE >= 250 THEN DO;
    PASSREC = 'A';
    CELL = '1HH';
  END;

  ELSE IF 215 <= SCORE <= 249 THEN DO;
    PASSREC = 'A';
    CELL = '4MH';
  END;

  ELSE DO;
    PASSREC = 'T';
    CELL = '7LH';
  END;
END;
OTHERWISE;
END;
END;

ELSE IF CRD_NAME = 'US1002D' THEN DO;

SELECT (BURSCR);
WHEN ('NONE') DO;
  CELL = 'NO BSCORE';
  IF SCORE >= 230 THEN PASSREC = 'A';
  ELSE IF 200 <= SCORE <= 229 THEN PASSREC = 'T';
  ELSE PASSREC = 'R';
END;

WHEN ('LOW') DO;
  IF SCORE >= 230 THEN DO;
    PASSREC = 'A';
    CELL = '3HL';
  END;

  ELSE IF 200 <= SCORE <= 229 THEN DO;
    PASSREC = 'T';
    CELL = '6ML';
  END;

  ELSE DO;
    PASSREC = 'R';
    CELL = '9LL';
  END;
END;

WHEN ('MED') DO;
  IF SCORE >= 230 THEN DO;
    PASSREC = 'A';
    CELL = '2HM';
  END;

  ELSE IF 200 <= SCORE <= 229 THEN DO;
    PASSREC = 'A';
    CELL = '5MM';
  END;

  ELSE DO;
    PASSREC = 'R';
    CELL = '8LM';
  END;

```

```

END;

WHEN ('HIGH') DO;
  IF SCORE >= 250 THEN DO;
    PASSREC = 'A';
    CELL = '1HH';
  END;

  ELSE IF 200 <= SCORE <= 229 THEN DO;
    PASSREC = 'A';
    CELL = '4MH';
  END;

  ELSE DO;
    PASSREC = 'T';
    CELL = '7LH';
  END;
END;
OTHERWISE;
END;
END;

*-----*;
RECREATE ACTUAL LOAN TO VALUE WITH ESTIMATED IF
NECESSARY
*-----*;
IF ACT_LTV IN (.,0) THEN ACT_LTV = EST_LTV;
*-----*;
CREATE TWO DATASETS, APPROVALS & DENIALS
*-----*;
DATA APPROVAL
  DENIAL(RENAME=(APPID   = D_APPID
                  CRD_NAME = D_CRDNAM
                  OR_USER  = D_ORUSER
                  OR_REASN = D_ORCODE
                  CELL     = D_CELL
                  ADJ_USER = D_ADJUSE
                  ADJ_CODE = D_ADJCDE
                  APRVPROD = D_APRVPR
                  APP_AMT  = D_APPAMT
                  CAT      = D_CAT
                  PASSREC  = D_PASSRE ));

LENGTH CATEGORY $30. CLASSIFY $11. ;
SET LAPS2;

*-----*;
CREATE COMPARISON CATEGORIES BASED ON
SCORECARD
*-----*;
*-----*;
CHANGING APPROVAL AMOUNT TO DENIAL APPROVAL
AMOUNT--REASON IS BASED ON AMOUNT DECLINED
NOT ON AMOUNT APPROVED
*-----*;
IF CRD_NAME = 'SE1002D' AND APP_AMT <= 50000
THEN DO;
  CAT = 'GROUP1';
  CATEGORY = 'HOME EQUITY, UP TO $50,000';
END;

```

```

ELSE IF CRD_NAME = 'SE1002D' AND APP_AMT > 50000
  THEN DO;
  CAT = 'GROUP2';
  CATEGORY = 'HOME EQUITY, ABOVE $50,000';
  END;

ELSE IF CRD_NAME = 'US1002D' AND APP_AMT <= 15000
  THEN DO;
  CAT = 'GROUP3';
  CATEGORY = 'UNSECURED LOANS UP TO $15,000';
  END;

ELSE IF CRD_NAME = 'US1002D' AND APP_AMT > 15000
  THEN DO;
  CAT = 'GROUP4';
  CATEGORY = 'UNSECURED LOANS ABOVE $15,000';
  END;

*-----*;
SET BOUNDARIES
*-----*;
LOW_AMT = APP_AMT * 0.9;
HIGH_AMT = APP_AMT * 1.1;

IF ADJ_CODE = 'A' THEN DO;
  IF PASSREC = 'A' THEN DO;
    IF CAT IN ('GROUP2' 'GROUP4')
      THEN CLASSIFY = 'INVESTIGATE';
    ELSE CLASSIFY = 'AUTO AP';
  END;

  ELSE IF PASSREC EQ 'T'
    THEN CLASSIFY = 'INVESTIGATE';
  ELSE IF PASSREC EQ 'R' AND COMPRESS(OR_USER)
    NOT IN ('', '-SYSTEM') THEN CLASSIFY = 'AUTO TD';

  OUTPUT APPROVAL;
  END;

ELSE IF ADJ_CODE EQ 'R' AND
  (STARTDTE <= DATE_REC <= LASTDATE) AND
  GENDER IN ('1', '2') AND
  RACE IN ('1', '2', '3', '4') AND
  PROCEED EQ 'Y' THEN DO;

IF PASSREC EQ 'A' AND COMPRESS(OR_USER) NOT IN
  ('', '-SYSTEM') THEN DO;
  IF CAT IN ('GROUP2' 'GROUP4')
    THEN CLASSIFY = 'INVESTIGATE';
  ELSE CLASSIFY = 'AUTO AP';
  END;

  ELSE IF PASSREC EQ 'T'
    THEN CLASSIFY = 'INVESTIGATE';
  ELSE IF PASSREC EQ 'R' THEN CLASSIFY = 'AUTO TD';

  OUTPUT DENIAL;
  END;

PROC PRINT DATA = DENIAL;
TITLE1 'DENIAL DATASET';

```

```

*-----*;
USE PROC DATASETS TO DELETE UNNECESSARY SAS
DATASETS AND MAKE MORE TEMPORARY SPACE
AVAILABLE.
*-----*;
PROC DATASETS LIBRARY=WORK;
  DELETE LAPS1 LAPS2 CREDIT CNTLACCT;

*-----*;
CREATE MATRIX
*-----*;
PROC SQL;
  CREATE VIEW PAIRS AS
  SELECT APPID, OR_USER, CRD_NAME,
         ADJ_CODE, OR_REASN, CELL,
         APRVPROD, APP_AMT,
         APPROVAL.CATEGORY,
         APPROVAL.HIGH_AMT,
         APPROVAL.CLASSIFY, CAT,
         PASSREC, ADJ_USER,
         APPROVAL.LOW_AMT,

         D_APPID, D_ORUSER, D_CRDNAM,
         D_ADJCDE, D_CELL, D_APRVPR,
         D_APPAMT, DENIAL.CATEGORY,
         D_ORCODE, DENIAL.CLASSIFY, D_CAT,
         D_PASSRE, DENIAL.D_TDCODE, D_ADJUSE

FROM APPROVAL, DENIAL
WHERE   APRVPROD = D_APRVPR           AND
        APPROVAL.CATEGORY = DENIAL.CATEGORY AND
        APPROVAL.CLASSIFY = DENIAL.CLASSIFY AND
        CELL = D_CELL               AND
        (D_APPAMT BETWEEN APPROVAL.LOW_AMT AND
        APPROVAL.HIGH_AMT);

DATA NEWPAIRS;
SET PAIRS;

PROC SORT DATA=NEWPAIRS;
  BY CATEGORY CLASSIFY;

*-----*;
GENERATE REPORT
*-----*;
PROC REPORT DATA = NEWPAIRS
  NOWINDOWS HEADLINE HEADSKIP MISSING;
  BY CATEGORY CLASSIFY;
  COLUMN D_APPID D_CELL D_ADJCDE D_ADJUSE
         D_TDCODE D_ORCODE D_ORUSER
         D_APPAMT
         APPID CELL ADJ_CODE ADJ_USER
         OR_REASN OR_USER APP_AMT;

OPTIONS PAGENO = 1 PS=60;

DEFINE D_APPID /
  WIDTH=12 ORDER 'TD/APPLICATION/ID';
DEFINE D_CELL /
  WIDTH=5 ORDER 'TD/CELL';
DEFINE D_ADJCDE /
  WIDTH=4 ORDER 'TD/DEC/CDE';

```

```

DEFINE D_ADJUSE /
  WIDTH=8 ORDER TD/DEC/USER';
DEFINE D_TDCODE /
  WIDTH=4 ORDER TD/CODE';
DEFINE D_ORCODE /
  WIDTH=4 ORDER 'TD/OVER/RIDE/CODE';
DEFINE D_ORUSER /
  WIDTH=6 ORDER 'OVER/RIDE/USER';
DEFINE D_APPAMT /
  WIDTH=8 ORDER FORMAT=DOLLAR8.
  TD/APPR/AMT';
DEFINE APPID /
  WIDTH=12 'AP/APPLICATION/ID';
DEFINE CELL /
  WIDTH=5 'AP/CELL';
DEFINE ADJ_CODE /
  WIDTH=4 'AP/DEC/CDE';
DEFINE ADJ_USER /
  WIDTH=8 'AP/DEC/USER';
DEFINE OR_REASN/
  WIDTH=5 'AP/OVER/RIDE/CODE';
DEFINE OR_USER /
  WIDTH=6 'OVER/RIDE/USER';
DEFINE APP_AMT /
  WIDTH=8 FORMAT=DOLLAR8. 'APPR/AMT';
BREAK AFTER D_APPID / SKIP;
%TTILES

```

CONCLUSION

Knowing that this report is a tool for quickly and effectively monitoring the decision-making process and fair lending practices, the code is always under constant scrutiny and being improved. We have used this technique in many other analyses as well. We have found that that this is a fairly simply way to bring data together where there is no common "BY" variable. We have also found that the PROC Report is a flexible method for displaying the information.

SAMPLE REPORT

*** CONFIDENTIAL: FOR INTERNAL USE ONLY ***

COMPARISON REPORTING FOR ALL BANKS
 BY PRODUCT, CELL AND DECISION CODE
 APPLICATIONS WERE RECEIVED SEPTEMBER 25, 1996

-----CATEGORY=HOME EQUITY, UP TO \$50,000 CLASSIFY=AUTO AP-----

| APPLICATION ID | DEC CELL | DEC CODE | DEC USER | TD RSN | OR CODE | OR USER | APPR AMT | APPLICATION ID | DEC CELL | DEC CODE | DEC USER | RIDE CODE | RIDE USER | APPR AMT |
|-------------------|-------------|-------------|-------------|-----------|------------|------------|-------------|-------------------|-------------|-------------|-------------|--------------|--------------|-------------|
| 123456789 | 1HH | R | 123 | III | AAA | NN888 | \$30,000 | 987654321 | 1HH | A | -SYSTEM | | | \$32,000 |

REFERENCES

SAS Institute Inc. (1990), SAS Guide to Macro Processing, Version 6, Second Edition, Cary, NC: SAS Institute Inc.

SAS, SAS/ACCESS, AND SAS/GRAPH are registered trademarks or trademarks of SAS Institute Inc. In the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Deborah J. Blair
 479 Cassatt Court
 West Chester, PA 19380

W. Lily Hadinoto
 17 Forestal Circle
 Newark, DE 19711