Paper 214-26

Reducing Bias in a Propensity Score Matched-Pair Sample Using Greedy Matching Techniques

Lori S. Parsons, Ovation Research Group, Seattle, WA

ABSTRACT

Matching members of a treatment group (cases) to members of a no treatment group (controls) is often used in observational studies to reduce bias and approximate a randomized trial. There is often a trade-off when matching cases to controls and two types of bias can be introduced. While trying to maximize exact matches, cases may be excluded due to incomplete matching. While trying to maximize cases, inexact matching may result. Bias is introduced by both incomplete matching and inexact matching. Propensity scores are being used in observational studies to reduce bias. It has been shown that matching on a propensity score can result in similar matched populations. This paper will describe how to reduce matched-pair bias caused by incomplete matching and inexact matching. Cases will be matched to controls on the propensity score using the presented matching algorithm. SAS/STAT® LOGISTIC procedure code will be given to create the propensity score. A user-written SAS® macro will be given to create a propensity score matched-pair sample using greedy matching techniques. The results of using the presented code, run on a large observational database of myocardial infarction patients, will be given as an example.

INTRODUCTION

Observational Studies

In large observational studies there are often significant differences between characteristics of a treatment group and a no treatment group. Such differences should not exist in a randomized trial. These differences must be adjusted for in order to reduce treatment selection bias and determine treatment effect. There are several methods to reduce the bias of these differences and make the two groups more similar. One method is to perform a case-control matched analysis.

Propensity Scores

SAS/STAT® allows users to perform multivariate logistic regression with the LOGISTIC procedure. PROC LOGISTIC options allow users to calculate and save the predicted probability of the dependent variable, the propensity score, for each observation in the data set. This single score (between 0 and 1) then represents the relationship between multiple characteristics and the dependent variable as a single characteristic. In the case of an observational study, the dependent variable might be a treatment group. The propensity score would then be the predicted probability of receiving the treatment. One score would be calculated for each patient in the study.

Propensity scores are being used in observational studies to reduce bias. Three techniques being used are subclassification on the propensity score, regression adjustment using the propensity score, and case-control matching on the propensity score. This paper focuses on case-control matching on the propensity score.

Matching Algorithms

There are basically two types of matching algorithms. One is an optimal match algorithm and the other is a greedy match algorithm. A greedy algorithm is frequently used to match cases to controls in observational studies. In a greedy algorithm, a set of X Cases is matched to a set of Y Controls in a set of X decisions. Once a match is made, the match is not reconsidered. That match is the best match currently available. In an optimal matching algorithm, previous matches are reconsidered before making the current match. The algorithm presented in this paper is a greedy algorithm. The presented algorithm also uses the nearest available pair matching method. The cases are ordered and sequentially matched to the nearest unmatched control. If more than one unmatched control matches to a case, the control is selected at random.

Matched-Pair Samples

In an observational study with X Cases and Y Controls (X < Y), a complete matched-pair sample contains all X Cases matched to a subset of X Controls. An incomplete matched pair sample contains <X matched pairs. In both cases, each control is selected at most once.

Good matched-pair samples contain both closely matched individual pairs and balanced case and control groups. A pair is closely matched if the distance between the case and the control is small. When a single covariate is used to match, the distance can be viewed as the absolute difference in the values. When several covariates are used, distances must be determined in more complex ways. When several covariates are represented as a single propensity score, the distance can more simply be viewed as the absolute difference in the propensity score of the case and the control. Matching on propensity score can create good matched-pairs. Matching on the propensity score can also balance case and control groups, or create covariate balance. It has been shown that a sample matched on propensity score will be similar for all the covariates that went into computing the propensity score.

SAS GREEDY 5→1 DIGIT MATCH MACRO

The SAS Macro presented here is an improvement to a macro previously presented by the author (SUGI 24 Proceedings, 2000). The purpose of making improvements was to increase the number of matched-pairs while at the same time improving the goodness of the individual matched-pairs. The original match macro paper described performing a 3-digit case-control match on propensity score and a separate 4-digit case-control match on propensity score. The 3-digit match picked up more matches, and thus reduced the bias due to incomplete matching. The 4-digit match picked up fewer, but better matched pairs, and thus reduced the bias due to inexact matching. It was up to the analyst to determine which match to use. More detailed information about the original matching macro can be found in the previous paper.

The improved macro presented here makes "best" matches first and "next-best" matches next, in a hierarchical sequence until no more matches can be made. Best matches are those with the highest digit match on propensity score. The algorithm proceeds sequentially to the lowest digit match on propensity score. Goodness of matched pairs is defined as those with the least absolute difference in matched propensity score.

The data presented here are from a large observational database of myocardial infarction patients. The cases (N = 2,402) received an Early Intervention. The controls (N = 17,735) did not. The controls are described in this example as the Conservative group. Characteristics of the original population can be found in Table 1.

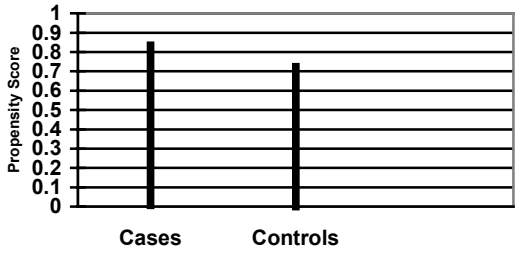
Appendix 1 contains the SAS PROC LOGISTIC code for the multivariate logistic regression that was run to compute the propensity scores and save the scores in a new data set. In this example, the propensity score is the predicted probability of having an Early Intervention. Appendix 1 also contains the macro call statement for the matching macro. More detailed information about creating a propensity in SAS can be found in the previous paper.

Appendix 2 contains the code for the SAS Greedy 5→1 Digit Match Macro. Greedy 5→1 Digit Match means that the cases were first matched to controls on 5 digits of the propensity score. For those that did not match, cases were then matched to controls on 4 digits of the propensity score. This continued down to a 1-digit match on propensity score for those that remained unmatched. Several variations of the same algorithm were performed and are described in Table 3. For this data, it was determined that the Greedy 5→1 Digit Match algorithm performed best based on completeness of match, goodness of matched sample, and goodness of matched pairs. It is therefore the one presented here.

INCOMPLETE MATCHES

Incomplete matching may result due to two reasons: they are missing data or disjoint ranges of case and control propensity scores. Data must be complete for all covariates in the multivariate analysis used to calculate the propensity score. If any covariate data is missing, the case is eliminated from the analysis and a propensity score is not calculated. Incomplete matching will result and the cases with missing data will be excluded. As displayed in Figure 1, the cases and the controls may contain a disjoint range of propensity scores. In the example data, the minimum and maximum propensity score for cases was 0.00683125 and 0.83533256. For controls, the minimum and maximum propensity score was 0.00103045 and 0.72406977. Incomplete matching will result and the cases with the highest propensity score (> 0.73 in the example data) and the controls with the lowest propensity score (<0.0068 in the example data) will be excluded.

Figure 1: Ranges of Propensity Scores



RESULTS OF MATCH

The results of running the logistic regression and the Greedy 5→1 Digit Match macro on the example data are given below. **Table 1** describes the original population and contains all covariates that were used in the multivariate logistic regression model to create the propensity score. Differences between groups were evaluated using the rank-sum test for continuous data and the chi-squared test for binary data. For every covariate, there was a significant difference between the cases (Early Intervention) and controls (Conservative) ($p < 0.05$).

Table 1: Original Population

	Early Intervention N (%)	Conservative N (%)	p-value
Total Patients	**2,402**	**17,735**	
Age (Mean±sd)	61.3±12.2	68.2±13.0	<0.0001
Male Gender	1,744 (72.6)	10,914 (61.5)	<0.0001
White Race	2,079 (91.8)	15,002 (88.4)	<0.0001
Hx Angina	444 (18.5)	4,441 (25.0)	<0.0001
Hx MI	574 (23.9)	5,382 (30.3)	<0.0001
Hx CHF	100 (4.2)	2,561 (14.4)	<0.0001
Hx CABG	378 (15.7)	3,312 (18.7)	0.0005
Hx PTCA	357 (14.9)	1,938 (10.9)	<0.0001
Hx Diabetes	394 (16.4)	4,895 (27.6)	<0.0001
Hx Smoking	816 (34.0)	4,638 (26.2)	<0.0001
Rales	226 (9.5)	2,956 (16.8)	<0.0001
Pulm. edema	73 (3.1)	1,437 (8.2)	<0.0001
Pulse > 100	285 (12.0)	4,404 (25.2)	<0.0001
Sys BP <=100	232 (9.7)	1,003 (5.7)	<0.0001
Chest Pain	2,180 (92.6)	14,164 (82.1)	<0.0001
Dx MI	1,040 (43.9)	2,598 (14.9)	<0.0001
Transferred	430 (17.9)	5,078 (28.6)	<0.0001

Table 2 describes the matched population based on the Greedy 5→1 Digit Match algorithm. For the matched analysis, differences between matched pairs were evaluated using the signed rank test for continuous data and the McNemar's test for binary data. For every covariate, there is no longer a significant difference between cases and controls. Eight-five percent of the cases were matched to a control. Of those that did not match, N = 242 (10%) did not match due to missing data and N = 124 (5%) did not match due to disjoint ranges of propensity scores.

Table 2: Greedy 5→1 Digit Match

	Early Intervention N (%)	Conservative N (%)	p-value
Total Patients	**2,036**	**2,036**	
Age (Mean±sd)	61.9 ± 12.0	61.7 ±13.3	0.5405
Male Gender	1,452 (71.3)	1,445 (71.0)	0.8087
White Race	1,865 (91.6)	1,858 (91.3)	0.6952
Hx Angina	390 (19.2)	381 (18.7)	0.7189
Hx MI	488 (24.0)	491 (24.1)	0.9124
Hx CHF	94 (4.6)	105 (5.2)	0.4240
Hx CABG	322 (15.8)	310 (15.2)	0.6036
Hx PTCA	292 (14.3)	264 (13.0)	0.2013
Hx Diabetes	345 (16.9)	355 (17.4)	0.6779
Hx Smoking	681 (33.4)	690 (33.9)	0.7654
Rales	191 (9.4)	193 (9.5)	0.8979
Pulm. edema	62 (3.0)	66 (3.2)	0.7143
Pulse > 100	254 (12.5)	257 (12.6)	0.8872
Sys BP <=100	180 (8.8)	197 (9.7)	0.3581
Chest Pain	1,877 (92.2)	1,872 (91.9)	0.7719
Dx MI	839 (41.2)	840 (41.3)	0.9746
Transferred	354 (17.4)	352 (17.3)	0.9340

EVALUATE MATCHES AND ALGORITHMS

It is up to the analyst to evaluate matched populations for completeness of match, goodness of matched sample and goodness of matched pairs. It is also up to the analyst to determine if a better match could be made. The macro presented here can be easily modified and variations of the same algorithm can be tried.

For the data presented here, several variations of the same algorithm were tried. **Table 3** describes the results. For the goodness of matched sample, only the mean ages of the cases and controls are shown in Table 3. The reason for this is to show that the goodness of the matched sample improves as the completeness of the match improves. Conversely, the goodness of the matched pairs decreases as the completeness of the match improves. All selected characteristics for the original population and the matched population are shown in Table 1 and Table 2.

The 4-digit match and 3-digit match were completed with the macro presented in the previous paper. The 5→3, 5→2, 5→1 and 6→1 digit matches were performed with the macro presented here. It can be seen from Table 3 that the macro presented here makes better matched pairs. The absolute difference in the propensity score of the 3-digit match was .00025 as compared to 0.00010 in the 5→3 Digit Match, while both algorithms matched the same number of cases (78%).

If the reservoir of controls is very large, the number of digits to select as a starting point could be increased. In the example presented here, a Greedy 6→1 Digit Match algorithm was also tried. The purpose was not to make more matches than the Greedy 5→1 Digit Match, as this algorithm matched the maximum possible, but to make better matched pairs. With this data, only N=76 cases matched to a control based on a 6→1 digit match. The absolute difference in propensity scores of matched pairs did not differ from the 5→1 digit match. Therefore, the 6→1 digit match was not an improvement over the 5→1 digit match.

The Greedy 5→1 Digit Match algorithm was selected to be the best algorithm for this data for the following reasons: 85% of the cases matched to a control and no additional matches could be made with the given data (completeness of the match); The p-value comparing cases to controls was not significant for any the selected criteria (goodness of the matched sample); The absolute difference in propensity score of matched pairs = 0.00043 (goodness of matched pairs); and the 6→1 digit match algorithm do not improve the match.

Table 3: Summary of Algorithms

	Completeness of Match	Goodness of Matched Sample		Goodness of Matched Pairs
Algorithm	Cases N (% Matched)	Cases Mean Age	Controls Mean Age	Absolute Difference in Propensity Score of Matched Pairs
Original Population	**2,402**	**61.3**	**68.2**	
4-Digit Match	1,405 (58.5%)	63.4	64.0	.000025
3-Digit Match	1,882 (78%)	62.3	62.2	.00025
5→3 Digit Match	1,882 (78%)	62.3	62.2	.00010
5→2 Digit Match	2,025 (84%)	61.9	61.7	.00035
5→1 Digit Match	**2,036 (85%)**	**61.9**	**61.7**	**.00043**
6→1 Digit Match	2,036 (85%)	61.9	61.7	.00043

CONCLUSIONS

The macro presented here, and variations of the macro presented here, can be used to perform a case-control match on propensity score. The match will be a good matched sample and contain good matched pairs. This can be used as a method to reduce selection bias in an observational study. A limitation to this method is the multivariate model from which the propensity score was computed. A poor model will result in a poor predicted probability of the outcome (propensity score). And, as with matching on individual characteristics, this method can only reduce bias in measured characteristics.

REFERENCES

D'Agostino, RB., Jr., "Tutorial in Biostatistics: Propensity Score Methods for Bias Reduction in the Comparison of a Treatment to a Non-Randomized Control Group", *Statistics in Medicine*, 1998, 17, 2265-2281.

Parsons, LS., "Using SAS® Software to Perform a Case-Control Match on Propensity Score in an Observational Study", *Proceedings of the Twenty-Fifth Annual SAS® Users Group International Conference,* Cary, NC: SAS Institute Inc., 2000; 1166-1171.

Rosenbaum, PR., "Optimal Matching for Observational Studies", *Journal of the American Statistical Association*, December 1989, 84:1024-1032.

Rosenbaum, P. and Ruben, D., "The Bias Due to Incomplete Matching", *Biometrics*, March 1985, 41, 103-116.

Rosenbaum, P. and Ruben, D., "The Central Role of the Propensity Score in Observational Studies for Causal Effects", *Biometrika*, 1983, 70, 41-55.

Rubin, DB., "Estimating Causal Effects from Large Data Sets Using Propensity Scores", *Annals of Internal Medicine*, October 1997, 127:757-763.

SAS Institute Inc. (1989), *SAS/STAT® User's Guide, Version 6, Fourth Edition*, Volume 1, Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

The example data presented here is from the National Registry of Myocardial Infarction (NRMI) database. The author wishes to thank Genentech, Inc. for use of the NRMI data.

TRADEMARKS

CONTACT INFORMATION

Contact the author at:

Lori S. Parsons
Ovation Research Group
305 Fir Place
Edmonds, WA 98020
Work Phone: (425) 672-8782
Fax: (425) 672-7282
Email: lparsons@ovation.org

APPENDIX 1:

```
/* ********************************  */
/* Perform a Logistic Regression and save*/
/* the propensity score data set         */
/* STUDY.Propen for all patients in the  */
/* observational study.                  */
/* Statistic Name = PROB                 */
/* Note:  PARMLABEL is a SAS Verion 8.0  */
/* option.                               */
/* ********************************** */

LIBNAME STUDY 'D:\INTERVEN';
PROC LOGISTIC DATA=STUDY.SEarly Descend;
MODEL interven = ptage male white
       mhprevmi mhangina mhchf mhptca
       mhcabg mhdiab mhsmoke killip1
       pulsecd2 bpsyscd2 admitmi cpcd
       tincd
  /SELECTION = STEPWISE RISKLIMITS
       LACKFIT RSQUARE PARMLABEL;
OUTPUT OUT= STUDY.Propen prob=prob ;
RUN;

/* ********************************** */
/* Call statement for Greedy Match Macro */
/* ********************************** */
%GREEDMTCH(STUDY,Propen,interven,Matches);
```

APPENDIX 2:

```
/* ********************************** */
/* Greedy 5->1 Digit Matching Macro      */
/* ********************************** */
%MACRO GREEDMTCH
 (
  Lib,        /* Library Name            */
  Dataset,    /* Data set of all         */
              /* patients                */
  depend,     /* Dependent variable      */
              /* that indicates          */
              /* Case or Control;        */
              /* Code 1 for Cases,       */
              /*      0 for Controls     */
  matches     /* Output file of matched  */
             /* pairs                    */
 );


/* Macro to sort the Cases and Controls
dataset */
%MACRO SORTCC;
proc sort data=tcases
   out=&LIB..Scase;
   by prob;
run;
proc sort data=tctrl
   out=&LIB..Scontrol;
   by prob randnum;
run;
%MEND SORTCC;

/* Macro to Create the initial Case and
Control Data Sets */
%MACRO INITCC(digits);
data tcases (drop=cprob)
     tctrl  (drop=aprob) ;
   set &LIB..&dataset. ;
   /* Create the data set of Controls*/
   if &depend. = 0 and prob ne .
   then do;
      cprob  = Round(prob,&digits.);
      Cmatch = 0;
      Length RandNum 8;
      RandNum=ranuni(1234567);
      Label RandNum=
         'Uniform Randomization Score';
      output tctrl;
    end;
   /* Create the data set of Cases   */
   else if &depend. = 1 and prob ne .
   then do;
      Cmatch = 0;
      aprob =Round(prob,&digits.);
      output tcases;
   end;
run;
%SORTCC;
%MEND INITCC;

/*  Macro to Perform the Match */
%MACRO MATCH (MATCHED,DIGITS);
data &lib..&matched. (drop=Cmatch randnum
aprob cprob start oldi curctrl matched);
   /* select the cases data set */
   set &lib..SCase ;
   curob + 1;
   matchto = curob;

   if curob = 1 then do;
      start = 1;
      oldi = 1;
   end;
   /* select the controls data set */
   DO i = start to n;
      set &lib..Scontrol point= i nobs = n;

      if i gt n then goto startovr;
      if _Error_ = 1 then abort;

      curctrl = i;
      /* output control if match found */
      if aprob = cprob then
      do;
         Cmatch = 1;
         output &lib..&matched.;
         matched = curctrl;
         goto found;
      end;
     /* exit do loop if out of potential
        matches */
      else if cprob gt aprob then
         goto nextcase;

    startovr: if i gt n then
       goto nextcase;
   END; /* end of DO LOOP  */
   /* If no match was found, put pointer
```

```
   back*/
   nextcase:
   if Cmatch=0 then start = oldi;
   /* If a match was found, output case and
      increment pointer */
   found:
   if Cmatch = 1 then do;
      oldi = matched + 1;
      start = matched + 1;
            set &lib..SCase point = curob;
         output &lib..&matched.;
   end;

   retain oldi start;
   if _Error_=1 then _Error_=0;
run;

/* Get files of unmatched cases and     */
/* controls.  Note that in the example  */
/* data, the patient identifiers are HID*/
/* (Hospital ID) and PATIENTN (Patient  */
/* identifier.  All cases have complete */
/* data for these two fields.  Modify   */
/* these fields with the appropriate    */
/* patient identifier field(s)          */
proc sort data=&lib..scase out=sumcase;
  by hid patientn;
run;
proc sort data=&lib..scontrol
out=sumcontrol;
  by hid patientn;
run;
proc sort data=&lib..&matched. out=smatched
(keep=hid patientn matchto);
   by hid patientn;
run;
data tcases (drop=matchto);
  merge sumcase(in=a) smatched;
  by hid patientn;
  if a and matchto = . ;
  cmatch = 0;
  aprob =Round(prob,&digits.);
run;
data tctrl (drop=matchto);
  merge sumcontrol(in=a) smatched;
  by hid patientn;
  if a and matchto = . ;
  cmatch = 0;
  cprob  = Round(prob,&digits.);
run;
%SORTCC
%MEND MATCH;

/* Note:  This section can be        */
/* modified to try variations of the */
/* basic algorithm.                  */
/* Create file of cases and controls */
%INITCC(.00001);
/* Do a 5-digit match */
%MATCH(Match5,.0001);
/* Do a 4-digit match on remaining
unmatched */
%MATCH(Match4,.001);
/* Do a 3-digit match on remaining
unmatched */
%MATCH(Match3,.01);
/* Do a 2-digit match on remaining
unmatched */
%MATCH(Match2,.1);
/* Do a 1-digit match on remaining
unmatched */
%MATCH(Match1,.1);

/* Merge all the matches into one file */
/* The purpose of the marchto variable */
/* is to identify matched pairs for the*/
/* matched pair anlayses.  matchto is  */
/* initially assigned the observation  */
/* number of the case.  Since there    */
/* would be duplicate numbers after the*/
/* individual files were merged,       */
/* matchto is incremented by file.     */
/* Note that if the controls file      */
/* contains more than N=100,000 records*/
/* and/or there are more than 1,000    */
/* matches made at each match level,   */
/* then the incrementation factor must */
/* be changed.                         */
data &lib..&matches.;
   set &lib..match5(in=a)
&lib..match4(in=b) &lib..match3(in=c)
&lib..match2(in=d) &lib..match1(in=e);
  if b then matchto=matchto + 100000;
  if c then matchto=matchto + 10000000;
  if d then matchto=matchto + 1000000000;
  if e then matchto=matchto + 100000000000;
run;
/* Sort file -- Need sort for Univariate
analysis in tables
                    */
proc sort data=&lib..&matches. out =
&lib..S&matches.;
   by &depend.;
run;

%MEND GREEDMTCH;
```