

Performing a 1:N Case-Control Match on Propensity Score

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

ABSTRACT

A case-control matched analysis is often used in observational studies to reduce selection bias and approximate a randomized trial. A propensity score is the predicted probability of an outcome. 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. Thus, matching on the propensity score can reduce the selection bias in an observational study.

This paper presents a macro that performs a 1:N case-control match on the propensity score; the analyst is allowed to specify the number of controls to match to each case. The macro makes "best" matches first and "next-best" matches next, in a hierarchical sequence until no more matches can be made. Each control is selected at most once. The final matched-pair samples contain both closely matched individual pairs and balanced case and control groups.

In this paper, SAS/STAT® LOGISTIC procedure code is given to create the propensity score. The matching macro is explained and used to create several propensity score matched-pair samples. The examples were run under SAS 8.2. Knowledge of logistic regression analysis and SAS macro language is required.

INTRODUCTION

Observational Studies

In large observational studies there are often significant differences between characteristics of the treatment group (cases) and the no treatment group (controls). 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 this type of bias and make the two groups more similar. One method is to perform a case-control matched analysis.

Propensity Scores

The SAS/STAT® software 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.

Propensity scores are being used in observational studies to reduce bias. Three commonly used techniques 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.

The 1:N Case-Control Matching Macro

The matching macro presented here (OneToManyMTCH) is an update to a macro previously presented by the author (SUGI 26 Proceedings, 2001). The original macro makes a 1:1 case-control match on the propensity score. It has been used to perform propensity score matched analyses for many published papers. The SAS macro presented here uses a similar algorithm, but is updated such that the user can specify the number of controls matched to each case (1:N). An individual control is picked at most one time.

Both the original 1:1 and the new 1:N matching macros are greedy algorithms; 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. The algorithm 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. First, cases are matched to controls on 8 digits of the propensity score. For those that do not match, cases are then matched to controls on 7 digits of the propensity score. The algorithm proceeds sequentially to the lowest digit match on propensity score (1 digit). This will be referred to as the 8→1 Digit Match.

In the 1:N matching macro presented here, all cases are initially matched to their "best" control in the first iteration of the 8→1 Digit Match. The set of matched cases is then matched to the set of un-matched controls in N-1 additional iterations of the 8→1 Digit Match. If a case does not have N matched controls, it is removed from the set of matches at the time it fails to receive a matched control. The corresponding control is also removed from the set of matches. The control is added back to the pool of un-matched controls, and allowed to match to another case.

METHOD

Example Data

The data presented here are from a large observational database of myocardial infarction patients. The cases (N=4,705) received treatment. The controls (N=15,212) did not receive treatment. **Table 1** describes the baseline characteristics of the original population. All of these baseline characteristics were independent variables 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 except race, there was a significant difference between the cases and controls ($p < 0.05$).

Create the Propensity Score

Below is the SAS PROC LOGISTIC code for the multivariate logistic regression that was run to compute the propensity scores; one score per patient. In this example, the propensity score is the predicted probability of having the treatment (**revasc**). The propensity score is saved in the file **study.AllPropen**.

```

/* Define the library for study data */
LIBNAME study
      'C:\Projects\SUGI_29\DataSetX';
/* ***** */
/* Perform the Logistic Regression */
/* Calculate and save propensity score */
/* Propensity score name = PROB */
/* Output file = STUDY.AllPropen */
/* ***** */
PROC LOGISTIC DATA = study.contra descend;
  MODEL revasc = ptage sex white mlrphecgr
    rwmisxhrr mhsmoke mndiab mhprevmi
    mhhyperc mhcad mhchf mhhypert mhptca
    mhcabg mhangina mhstroke renalcd
    copdcd mlrcshkcd prbpsys prbpdia
    prpulse
  / SELECTION = STEPWISE RISKLIMITS
    LACKFIT RSQUARE PARMLABEL;
  OUTPUT OUT=study.AllPropen prob=prob ;
RUN;

```

Table 1: Original Population			
	Treatment N (%)	No Treatment N (%)	p-value
Total Patients	4,705	15,212	
Age Mean \pm sd	65.9 \pm 12.84	73.3 \pm 12.44	<0.0001
Female	1,703 (36.2)	7,169 (47.1)	<0.0001
White Race	3,934 (84.1)	12,824 (84.7)	<0.3701
ECG Obtained	406 (8.8)	724 (4.9)	<0.0001
Sx to Door(hr) Mean \pm sd	2.1 \pm 1.9	2.6 \pm 2.7	<0.0001
Hx Angina	573 (23.2)	2,342 (15.4)	<0.0001
Hx MI	1,023 (21.7)	4,584 (30.1)	<0.0001
Hx CHF	288 (6.1)	3,304 (21.7)	<0.0001
Hx CABG	373 (7.9)	2,254 (14.8)	<0.0001
Hx PTCA	663 (14.1)	1,195 (7.9)	<0.0001
Hx Diabetes	1,069 (22.7)	4,767 (31.3)	<0.0001
Hx Smoking	1,447 (31.6)	2,822 (19.0)	<0.0001
Hx Hyperchol.	1,428 (30.4)	3,276 (21.5)	<0.0001
Hx CAD	1,210 (25.7)	2,872 (18.9)	<0.0001
Hx Hyperten.	2,787 (59.2)	9,366 (61.6)	0.0041
Hx Stroke	1,845 (39.2)	5,392 (35.4)	<0.0001
Hx Renal	131 (5.3)	1,180 (13.7)	<0.0001
Hx COPD	288 (11.6)	1,397 (16.3)	<0.0001
Chest Pain	4,098 (89.2)	8,998 (61.8)	<0.0001
Killip Class IV	271 (5.8)	752 (4.9)	0.0266
Systolic BP Mean \pm sd	138.8 \pm 35.9	140.0 \pm 38.1	0.0011
Diastolic BP Mean \pm sd	80.7 \pm 20.7	79.7 \pm 20.9	0.0035
Pulse Mean \pm sd	80.2 \pm 23.1	92.1 \pm 26.6	<0.0001
Propensity Score N (Mean \pm sd) (min, max)	4,416 .3990 \pm .1960 .0064, .9285	13,931 .1905 \pm .1667 .0011, .8780	

Include the 1:N Matching Macro

Appendix 1 contains the matching macro OneToManyMTCH code. Include the macro into a SAS session as follows:

```

/* Include the matching macro code */
%include 'C:\Projects\SUGI_29\SASPrograms\MacroOneToManyMatch.txt';

```

1:N Match Where N = 1

Figure 1 displays what the matching macro does in the initial 1:1 match iteration. The numbers of records at each step are from the 1:1 match of the example data.

Step A: The macro creates separate files for the cases and controls.

Step B: Next, it performs the initial 8 \rightarrow 1 Digit Match.

Step C: It then outputs a file of the matched pairs. This file contains the field to link the matched pairs (MATCH_1).

Step D: Finally, it outputs a file of un-matched controls. This will be the pool from which the matches are made in the next iteration.

1:N Match Where N = 2 to N

Figure 2 displays what the matching macro does for the remaining iterations. The numbers of records at each step are from the 1:2 match of the example data.

Step A: The macro starts with the previously matched cases and pool of un-matched controls.

Step B: It then performs another 8 → 1 Digit Match.

Step C: Next, it outputs a file of new matched pairs. This file also contains the field to link this set of matched pairs (MATCH_X where X is the number of the iteration). Note that any cases from a previous iteration, that did not receive a match in the current iteration, are no longer in the current set of matched pairs.

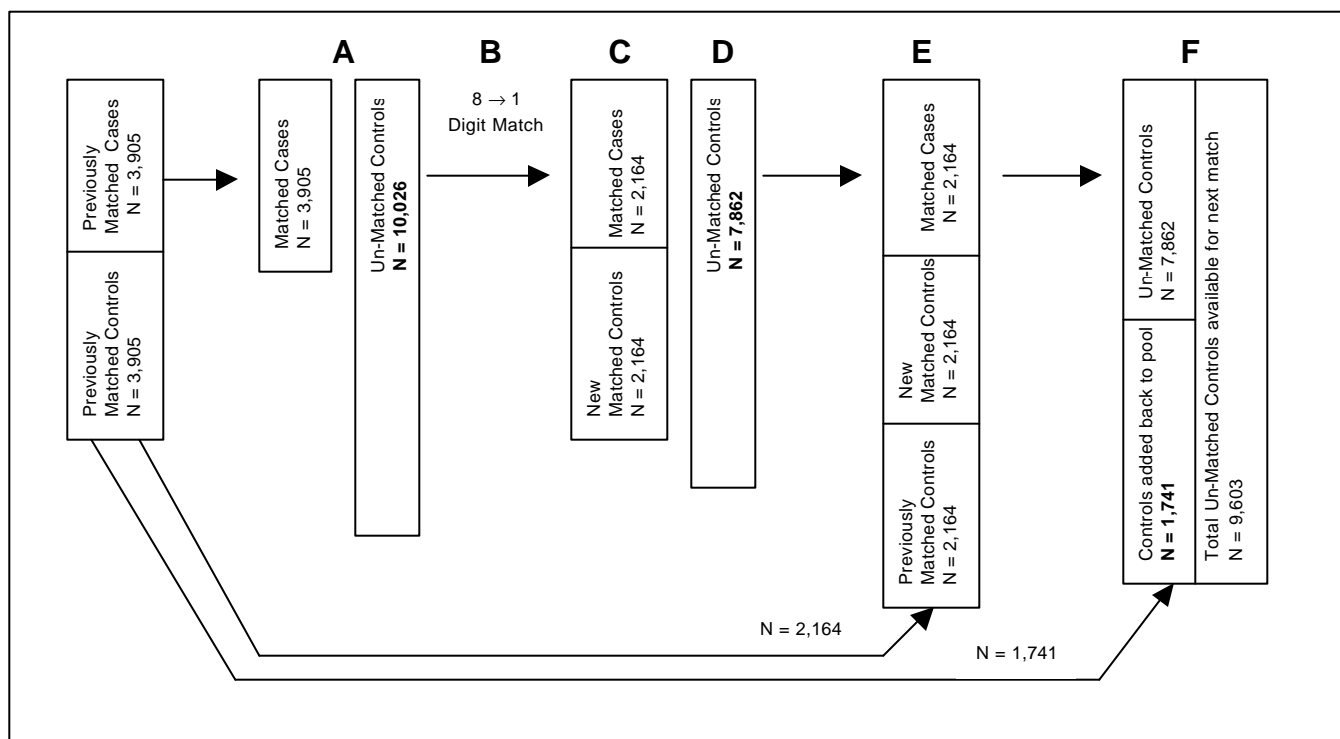
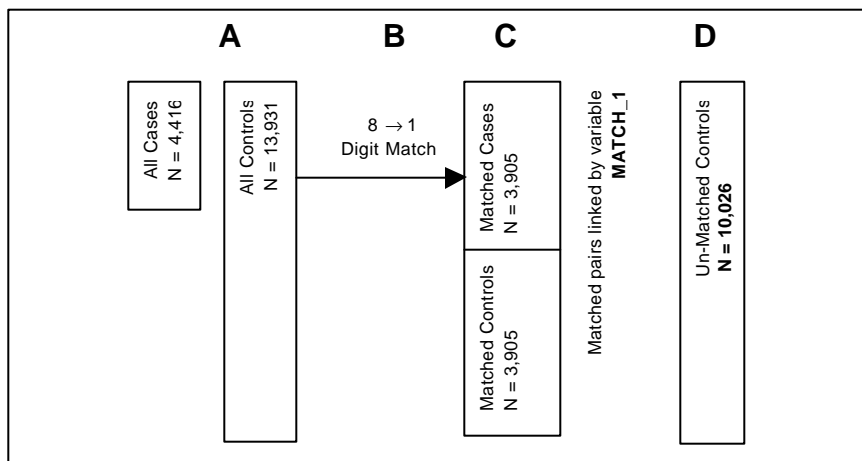
Step D: The macro then outputs the file of un-matched controls.

Step E: Next it determines which controls matched to a case in a previous iteration. If that case still exists in the current set of matched pairs, the macro appends the control to the file of current matched pairs.

Step F: Finally, it determines which controls matched to a case in a previous iteration, but whose matched case did not receive a match in the current iteration. These controls are appended to the file of un-matched controls. Thus, these controls are added back to the pool of available controls to be potential matches in the next iteration.

Figure 2: 1:N Match Where N = 2 to N

Figure 1: 1:N Match Where N = 1



Call the 1:N Matching Macro

The call variables to the macro are described below in the comments.

```
%MACRO OneToManyMTCH (
  Lib,          /* Library containing the propensity score data */
  Dataset,     /* Data set containing the propensity score */
  Depend,      /* Dependent variable that indicates Case or Control*/
              /* Code 1 for Cases, 0 for Controls */
  SiteN,       /* Site/Hospital ID */
  PatientN,    /* Patient ID */
  Matches,     /* Output data set of matched pairs */
  NoContrl;    /* Number of controls to match to each case */
);
```

The following are examples of call statements to the macro. The first is for a 1:1 match, the second is for a 1:2 match and the third is for a 1:3 match.

```
/* Call Macro and Perform 1:1 Match */
%OneToManyMTCH(STUDY,ALLPropen,revasc,hid,patientn,Matches_1,1);

/* Call Macro and Perform 1:2 Match */
%OneToManyMTCH(STUDY,ALLPropen,revasc,hid,patientn,Matches_2,2);

/* Call Macro and Perform 1:3 Match */
%OneToManyMTCH(STUDY,ALLPropen,revasc,hid,patientn,Matches_3,3);
```

The Final File of Matched Cases and Controls

The final output data set will contain all the matched pairs and a series of fields to link the pairs. The fields will be named Match_1, Match_2, ... Match_N. The matched case-control pair made during the first match can be linked with the Match_1 variable (the variables will have the same numeric value). The matched case-control made during the second match can be linked with the Match_2 variable, and so on.

The number of matches per digit and the number of matches at each match level can be determined by looking at the SAS Log (**Figure 3**). The file MATCH8 includes the matches made on 8 digits of the propensity score. The file MATCH7 includes the matches made on 7 digits of the propensity score, and so on. The last number of the merged file name will denote which match was made. In the following log, the merged file is named ALLMATCHES_31. This file was created after the 1:1 match.

Figure 3: SAS Log of Files Created During 1:1 Match

Log
NOTE: There were 2 observations read from the data set STUDY.MATCH8.
NOTE: There were 8 observations read from the data set STUDY.MATCH7.
NOTE: There were 116 observations read from the data set STUDY.MATCH6.
NOTE: There were 1068 observations read from the data set STUDY.MATCH5.
NOTE: There were 3676 observations read from the data set STUDY.MATCH4.
NOTE: There were 2354 observations read from the data set STUDY.MATCH3.
NOTE: There were 550 observations read from the data set STUDY.MATCH2.
NOTE: There were 36 observations read from the data set STUDY.MATCH1.
NOTE: The data set WORK.ALLMATCHES_31 has 7810 observations and 335 variables.

Incomplete Matches

Incomplete matching may result due to missing data, disjoint ranges of case and control propensity scores, and failure to match on the specified number of digits. 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. Cases and controls may contain a disjoint range of propensity scores. In the example data, the minimum and maximum propensity score for cases is 0.0064263 and 0.9285301. For controls, the minimum and maximum is 0.0011034 and 0.8780113. The cases with the highest propensity score (> 0.8780113) and the controls with the lowest propensity score (<0.0064263) will be excluded due to disjoint ranges.

RESULTS

Evaluate the Matches

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 modified and variations of the same algorithm can be tried.

If the reservoir of controls is very large, the number of match-digits to select as a starting point could be increased; the macro could be modified to start with a >8 digit match. To select only the best matches, the 1-digit match could be removed, making the algorithm an 8→2 digit matching algorithm. Or, if the reservoir of controls is smaller, the analyst could start with a <8-digit match.

Matched Samples

The results of performing a 1:1, 1:2 and 1:3 case-control match on the example data are given below. 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. With the exception of age (1:2 and 1:3 matched populations), history of stroke (1:2 and 1:3 matched populations), and history of smoking (1:3 matched population), there is no longer a significant difference between the characteristics of the cases and controls. **Table 2** describes the 1:1 matched population, **Table 3** describes the 1:2 matched population, and **Table 4** describes the 1:3 matched population.

Table 2: 1:1 Matched Population			
	Treatment N (%)	No Treatment N (%)	p-value
Total Patients	3,905	3,905	
Age Mean ± sd	67.1± 12.3	66.8 ± 13.7	0.5596
Female	1,481 (37.9)	1,468 (37.6)	0.7616
White Race	3,283 (84.1)	3,259 (83.5)	0.4615
ECG Obtained	300 (7.9)	278 (7.2)	0.2820
Sx to Door(hr) Mean ± sd	2.2 ± 2.0	2.2 ± 2.2	0.1327
Hx Angina	483 (12.4)	479 (12.3)	0.8905
Hx MI	872 (22.3)	888 (22.7)	0.6648
Hx CHF	259 (6.6)	243 (6.2)	0.4604
Hx CABG	340 (8.7)	331 (8.5)	0.7163
Hx PTCA	511 (13.1)	506 (13.0)	0.8665
Hx Diabetes	931 (23.8)	931 (23.8)	0.7505
Hx Smoking	1,150 (29.4)	1,122 (28.7)	0.4855
Hx Hyperchol.	1,116 (28.6)	1,110 (28.4)	0.8805
Hx CAD	963 (24.7)	950 (24.3)	0.7323
Hx Hyperten.	2,340 (59.9)	2,388 (61.2)	0.2665
Hx Stroke	1,475 (37.8)	1,454 (37.2)	0.6236
Hx Renal	118 (5.9)	125 (6.1)	0.8311
Hx COPD	243 (12.2)	231 (11.2)	0.3439
Chest Pain	3,401 (89.1)	3,411 (88.6)	0.4846
Killip Class IV	183 (4.7)	187 (4.8)	0.8313
Systolic BP Mean ± sd	139.8 ± 35.8	141.0 ± 37.0	0.8313
Diastolic BP Mean ± sd	80.7 ± 20.5	81.2 ± 20.9	0.5005
Pulse Mean ± sd	81.5 ± 23.3	82.0 ± 23.0	0.1378
Propensity Score Mean ± sd min, max	.3650± .1800 .0064, .8833	.3649± .1801 .0064, .8780	

Table 3: 1:2 Matched Population			
	Treatment N (%)	No Treatment N (%)	p-value
Total Patients	2,164	4,328	
Age Mean ± sd	71.7±10.9	72.5±11.7	0.0001
Female	976 (45.1)	1,946 (44.96)	0.9157
White Race	1,833 (84.7)	3,692 (85.3)	0.5216
ECG Obtained	104 (4.8)	209 (4.8)	0.9673
Sx to Door(hr) Mean ± sd	2.6 ± 2.2	2.6 ± 2.4	0.1713
Hx Angina	312 (14.4)	616 (14.2)	0.8410
Hx MI	578 (26.7)	1,138 (26.3)	0.7202
Hx CHF	237 (11.0)	472 (10.9)	0.9551
Hx CABG	281 (13.0)	540 (12.5)	0.5613
Hx PTCA	213 (9.8)	369 (8.5)	0.0799
Hx Diabetes	634 (29.3)	1,216 (28.1)	0.3120
Hx Smoking	470 (21.7)	874 (20.2)	0.1528
Hx Hyperchol.	518 (23.9)	956 (22.1)	0.0938
Hx CAD	462 (21.4)	892 (20.6)	0.4895
Hx Hyperten.	1,350 (72.4)	2,691 (62.2)	0.8706
Hx Stroke	766 (35.4)	1,685 (38.9)	0.0056
Hx Renal	106 (4.9)	201 (4.6)	0.6492
Hx COPD	155 (7.2)	337 (7.8)	0.3706
Chest Pain	1,683 (77.8)	3,426 (79.2)	0.1984
Killip Class IV	107 (4.9)	175 (4.0)	0.0931
Systolic BP Mean ± sd	142.4 ± 36.9	142.1 ± 36.5	0.2567
Diastolic BP Mean ± sd	80.4 ± 20.3	80.1 ± 19.8	0.5124
Pulse Mean ± sd	87.2 ± 25.0	86.8 ± 23.6	0.2776
Propensity Score Mean ± sd min, max	.2330± .1024 .0064, .4477	.2329± .1023 .0064, .4485	

Table 4: 1:3 Matched Population

	Treatment N (%)	No Treatment N (%)	p-value
Total Patients	1,767	5,301	
Age Mean \pm sd	72.0 \pm 10.8	73.0 \pm 11.7	<0.0001
Female	820 (46.4)	2,400 (45.3)	0.4081
White Race	1,494 (84.6)	4,530 (85.5)	0.3529
ECG Obtained	81 (4.6)	254 (4.8)	0.7222
Sx to Door(hr) Mean \pm sd	2.6 \pm 2.2	2.6 \pm 2.5	0.1714
Hx Angina	262 (14.8)	805 (15.2)	0.7155
Hx MI	488 (27.6)	1,443 (27.2)	0.7462
Hx CHF	221 (12.5)	652 (12.3)	0.8184
Hx CABG	245 (13.9)	688 (13.0)	0.3404
Hx PTCA	169 (9.6)	440 (8.3)	0.1011
Hx Diabetes	521 (29.5)	1,520 (28.7)	0.5147
Hx Smoking	394 (22.3)	1,049 (19.8)	0.0235
Hx Hyperchol.	414 (23.4)	1,184 (22.3)	0.3410
Hx CAD	364 (20.6)	1,061 (20.0)	0.5957
Hx Hyperten.	1,100 (62.3)	3,320 (62.6)	0.7766
Hx Stroke	608 (34.4)	2,020 (38.1)	0.0050
Hx Renal	94 (5.3)	249 (4.7)	0.2916
Hx COPD	131 (7.4)	425 (8.0)	0.4144
Chest Pain	1,318 (74.6)	3,996 (75.4)	0.5043
Killip Class IV	87 (4.9)	223 (4.2)	0.2026
Systolic BP Mean \pm sd	142.5 \pm 37.2	141.5 \pm 36.9	0.9093
Diastolic BP Mean \pm sd	80.3 \pm 20.4	79.7 \pm 19.9	0.6039
Pulse Mean \pm sd	88.8 \pm 25.6	87.9 \pm 24.1	0.6204
Propensity Score Mean \pm sd min, max	.2143 \pm .1039 .0064, .4477	.2155 \pm .1059 .0063, .4485	

CONCLUSIONS

The macro presented here will allow an analyst to perform a 1:N case-control match on propensity score. The match will be a well-matched sample and contain well-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., "Reducing Bias in a Propensity Score Matched-Pair Sample Using Greedy Matching Techniques". *Proceedings of the Twenty-Sixth Annual SAS[®] Users Group International Conference*, Cary, NC: SAS Institute Inc., 2001; <http://www2.sas.com/proceedings/sugi26/p214-26.pdf>.

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 are from the National Registry of Myocardial Infarction (NRMII) database. The author wishes to thank Genentech, Inc. for use of the NRMII data.

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: lparkers@ovation.org

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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

APPENDIX 1:

```

/* ***** */
/* ***** */
/* Matching Macro */
/* ***** */
/* ***** */
%MACRO OneToManyMTCH (
  Lib,          /* Library Name */
  Dataset,     /* Data set of all patients */
  depend,      /* Dependent variable that indicates Case or Control */
               /* Code 1 for Cases, 0 for Controls */
  SiteN,       /* Site/Hospital ID */
  PatientN,    /* Patient ID */
  matches,     /* Output data set of matched pairs */
  NoCtrls);   /* Number of controls to match to each case */

/* ***** */
/* Macro to Create the Case and Control Data sets */
/* ***** */
%MACRO INITCC(CaseAndCtrls,digits);
data tcases (drop=cprob)
  tctrl (drop=aprob) ;
set &CaseAndCtrls. ;
/* 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 sort the Cases and Controls data set */
/* ***** */
%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 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 */
proc sort data=&lib..scase out=sumcase;
  by &SiteN. &PatientN.;
run;
proc sort data=&lib..scontrol out=sumcontrol;
  by &SiteN. &PatientN.;
run;
proc sort data=&lib..&matched. out=smatched (keep=&SiteN. &PatientN. matchto);
  by &SiteN. &PatientN.;
run;

data tcases (drop=matchto);
  merge sumcase(in=a) smatched;
  by &SiteN. &PatientN.;
  if a and matchto = . ;
  cmatch = 0;
  aprob =Round(prob,&digits.);
run;
data tctrl (drop=matchto);
  merge sumcontrol(in=a) smatched;

```



```

by &SiteN. &PatientN.;

if a and matchto = . ;
  cmatch = 0;
  cprob = Round(prob,&digits.);
run;
%SORTCC

%MEND MATCH;

/* ***** */
/* Macro to call Macro MATCH for each of the 8-digit to 1-digit matchs */
/* ***** */
%MACRO CallMATCH;
  /* Do a 8-digit match */
  %MATCH(Match8,.0000001);
  /* Do a 7-digit match on remaining unmatched*/
  %MATCH(Match7,.000001);
  /* Do a 6-digit match on remaining unmatched*/
  %MATCH(Match6,.00001);
  /* Do a 5-digit match on remaining unmatched*/
  %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);

%MEND CallMATCH;

/* ***** */
/* Macro to Merge all the matches files into one file */
/* ***** */
%MACRO MergeFiles(MatchNo);
  data &matches.&MatchNo. (drop = matchto);
    set &lib..match8(in=a) &lib..match7(in=b) &lib..match6(in=c) &lib..match5(in=d)
&lib..match4(in=e)
      &lib..match3(in=f) &lib..match2(in=g) &lib..match1(in=h);
    if a then match &MatchNo. = matchto;
    if b then match_&MatchNo. = matchto + 10000;
    if c then match &MatchNo. = matchto + 100000;
    if d then match_&MatchNo. = matchto + 1000000;
    if e then match &MatchNo. = matchto + 10000000;
    if f then match_&MatchNo. = matchto + 100000000;
    if g then match_&MatchNo. = matchto + 1000000000;
    if h then match_&MatchNo. = matchto + 10000000000;
  run;
%MEND MergeFiles;

/* ***** */
/* ***** */
/* Perform the initial 1:1 Match */
/* ***** */
/* ***** */

/* Create file of cases and controls */
%INITCC(&LIB..&dataset.,.00000001);

/* Perform the 8-digit to 1-digit matches */
%CallMATCH;

/* Merge all the matches files into one file */
%MergeFiles(1)

```

```

/* ***** */
/* ***** */
/* Perform the remaining 1:N Matches */
/* ***** */
/* ***** */
%IF &NoCtrls. gt 1 %Then %DO;
  %DO i = 2 %TO &NoCtrls.;

    %let Lasti=%eval(&i. - 1);

    /* ***** */
    /* Start with Cases from the last Matched Cases file and the remaining Un-Matched */
    /* Controls. NOTE: The Unmatched Controls file (Scontrol) is created at end of the */
    /* previous match */
    /*
    /* Select the Matched Cases from the last Matched File */
    data &LIB..Scase;
      set &matches.&Lasti.;
      where &Depend. = 1;
    run;

    /* ***** */
    /* Perform the 8-1 digit matches between Matched Cases and the Unmatched Controls */
    %CallMATCH;

    /* ***** */
    /* Merge the 8-digit to 1-digit matches files into one file */
    %MergeFiles(&i.)

    %DO m = 1 %TO &Lasti.;
      data &matches.&i.;
        set &matches.&i.;
        if &Depend.=0 then Match_&m. = .;
      run;
    %END;

    /* ***** */
    /* Determine which OLD Controls correspond to the kept Cases */

    %DO c = 1 %TO &Lasti.;
      /* Select the KEPT Cases */
      proc sort data=&matches.&i. out=skeepcases (keep = Match_&c.);
        by Match_&c.;
        where &Depend. = 1;
      run;
      /* Get the OLD Controls */
      proc sort data = &matches.&Lasti. out = soldcontrols&c.;
        by Match_&c.;
        where &Depend. = 0 and Match_&c. ne . ;
      run;
      /* Get the OLD Controls that correspond to the kept Cases */
      data keepcontrols&c.;
        merge skeepcases (in = a) soldcontrols&c. (in = b);
        by Match_&c.;
        if a;
      run;
    %END;

    /* ***** */
    /* Combine all the OLD Controls into one file */

    data keepcontrols;
      set keepcontrols1 (obs=0);
    run;
    %DO k = 1 %TO &Lasti.;
      data keepcontrols;
        set keepcontrols keepcontrols&k.;

```

```

run;
%END;

/* ***** */
/* Append the OLD matched Controls to the new file of matched cases and controls */
data &matches.&i.;
set &matches.&i. keepcontrols;
run;

/* ***** */
/* If there are more matches to be made, add the previously matched, but not kept, */
/* controls back into the pool of unmatched controls */
%if &i. lt &NoCtrls. %then %do;

%DO z = 1 %TO &Lasti.;

/* Select all the KEPT Cases */
proc sort data=&matches.&i. out=skeepcases (keep = Match_&z.);
by Match &z.;
where &Depend. = 1;
run;

/* Select all the OLD Controls */
proc sort data = &matches.&Lasti. out = soldcontrols&z.;
by Match &z.;
where &Depend. = 0 and Match_&z. ne .;
run;

/* Keep the OLD Controls that correspond to the NOT KEPT Cases */
/* Drop the previous Match_X variable */
data AddBackControls&z. (drop = Match_&z.);
merge skeepcases (in = a) soldcontrols&z. (in = b);
by Match_&z.;
if b and not a;
run;
%END; /* End DO */

/* Drop the previous Match X variable */
data &LIB..Scontrol (drop = Match_&lasti. );
set &LIB..Scontrol;
run;

/* Append */
%DO y = 1 %TO &Lasti.;
data &LIB..Scontrol;
set &LIB..Scontrol AddBackControls&y.;
run;
%END; /* End DO */
%end; /* End IF */

%END; /* End Main DO */
%END; /* End Main IF */

/* ***** */
/* ***** */
/* Save the final matched pairs data set */
/* ***** */
/* ***** */
/* Sort file by Treatment Variable */

proc sort data=&matches.&NoCtrls. out = &lib..&matches.;
by &depend.;
run;

%MEND OneToManyMTCH;

```