Paper 230-25

Automating the Selection of Controls in Case-Control Studies

Edgar L. Mounib
Thiru Satchi
Blue Cross and Blue Shield of Massachusetts, Boston, Massachusetts

Abstract

A case-control study is an observational study in which subjects are classified according to the presence ('cases') or absence ('controls') of the outcome of interest. These controls can then be matched with cases according to a characteristic(s), to ensure the two groups are comparable for further analysis. However, this matching process can be difficult, time consuming, and costly. Hence, we developed a SAS® program to automate the selection of controls, taking into account age, gender, and other user-defined criteria. This program can be adjusted to account for varying case-control ratios, thus affecting the power of the study. It is intended for SAS programmers with a basic understanding of macro programming.

Keywords: case-control study, matching, macro, CALL EXECUTE

Introduction

A case-control study is an observational investigation used to measure the association between an outcome and the exposures and attributes suspected of causing (or preventing) that outcome. The key design element of a case-control study is the selection of subjects. Specifically, subjects are conditionally enrolled due to the presence ('cases') or absence ('controls') of the outcome, for the purpose of evaluating the relative frequency of exposures and attributes in those with and without the outcome. The case-control study design is commonly used in clinical and epidemiologic studies for several reasons. It allows for the study of rare and chronic diseases. Case-control investigations also tend to require a smaller sample, cost less to conduct, and are typically faster to complete than other observational studies .[1]

A primary concern of investigators who design case-control studies is confounding – a distortion of the effect of the exposure of interest due to the effects of an extraneous variable(s). Matching controls to cases on the basis of established risk factors for the outcome of interest is a common practice in case-control investigations. This forces comparability between these two groups with respect to these factors and thereby makes comparisons between them less susceptible to confounding. In addition, when the number of cases is limited or fixed, the statistical power of the study can be increased by matching more than one control per case (R:1 matching). However, the matching process can be difficult, time consuming, and costly.[1]

A previous presentation introduced a program that matches cases and controls.[2] However, this matching occurs for a single variable and on a one-to-one basis. We will now introduce the Case-Control Matching Program that features the following:

- matches on multiple variables and variable types, such as distinct (e.g., male versus female) and ranges (e.g., age ± 10 years) of values;
- randomizes the selection of controls; and
- accounts for an adjustable number of controls matched per case.

Case-Control Matching Program

The Case-Control Matching Program is located in Appendix A and illustrated in Appendix B. Each step of this program will now be explained. It should be noted that the 'Listings' in this section include words in bold. These words represent variables that can be modified by the user for his or her own personal use. And for the purpose of introducing this program, we will use two matching variables – age (±10 years) and gender – throughout the Listings.

Step 1

The first step of this program is to establish two global macro variables: "AGERANGE" and "RATIO" (Listing 1). AGERANGE refers to the allowable age range for the controls, relative to the age of each case. For example, if a case were 35 years old and we wanted to match this person to a control(s) who is either 10 years younger or older (i.e., 25 to 45 years), then we would enter 10 for this value. RATIO represents the desired number of controls matched to a single case.

Listing 1. Step 1 of the Case-Control Matching Program.

```
%LET AGERANGE = 10;
%LET RATIO = 5;
```

Step 2

The next stage is to separate the cases and controls from one another into separate data sets: "CASES" and "CONTROLS", respectively (Listing 2). For the purpose of this presentation, it is assumed both groups are present in a single data set and are differentiated by the variable, "CASECTRL" ('1' represents a case, '0' a control).

Listing 2. Step 2 of the Case-Control Matching Program.

```
DATA CASES CONTROLS;
  SET RAW_DATA;

  IF CASECTRL = 1 THEN OUTPUT CASES;
    ELSE OUTPUT CONTROLS;
```

Step 3

The third step of the program is two fold: (1) create a unique list of possible combinations of the matching values present in the case population and (2) calculate the frequency of these combinations in this population ('COUNT') (Listing 3). This is accomplished with PROC FREQ, an example of which is presented in Listing 3. Therefore, the list of possible combinations is reflected in the TABLES statement and is outputted to the temporary data set called 'CASEOUT'. This list will be used as the matching criteria in the next step.

Listing 3. Step 3 of the Case-Control Matching Program.

```
PROC FREQ NOPRINT DATA=CASE;
  TABLES AGE*GENDER/OUT=CASEOUT;
```

Step 4

The selection of the matched controls using a macro occurs in the next step (Listing 4). The first line of this macro ('SAMPLE') establishes the local macro variables used to store values from each combination of values identified in the third step of this program (i.e., 'V_AGE' represents the case's age and 'V_SEX' represents the case's gender). The 'V_COUNT' variable will store the number of times this combination occurs in the case population.

A subset of the control population, 'QUALIFY1', who meet the matching criteria, is created (Listing 4, Lines 2-4). User-defined variables referring to the matching variables are then established (Lines 6-7). These variables capture the matching criteria and will be carried forward for auditing purposes (refer to Step 5). The QUALIFY1 subset of controls is then assigned a random number ('SEED') and sorted by this number (Lines 9-10).

Listing 4. Step 4 of the Case-Control Matching Program.

```
1.  %MACRO SAMPLE(V_AGE,V_SEX,V_COUNT);
2.  DATA QUALIFY1; SET CONTROLS;
3.    WHERE (&V_AGE-&AGERANGE
        <=AGE<=&V_AGE+&AGERANGE)
4.      AND (GENDER = "&V_SEX");
5.
6.  CASE_AGE=&V_AGE;
7.  CASE_SEX="&V_SEX";
8.
9.  SEED=RANUNI(0);
10. PROC SORT;  BY SEED;
11.
12. DATA QUALIFY2;
13.   SET QUALIFY1 NOBS=TOTOBS;
14.     IF _N_ <= &V_COUNT*&RATIO;
15.     IF &VCNT*&RATIO <= TOTOBS THEN TAG = 'YES';
16.       ELSE TAG = 'NO';
17.
18. PROC APPEND BASE=MATCHES DATA=QUALIFY2;
19.
20. PROC SORT DATA=QUALIFY2 OUT=TEMP1
    (KEEP=UNIQUEID); BY UNIQUEID;
21.
22. PROC SORT DATA=CONTROL OUT=TEMP2;
    BY UNIQUEID;
23.
24. DATA CONTROL;
25.   MERGE TEMP1(IN=IN1) TEMP2(IN=IN2);
26.     BY UNIQUEID; IF IN2 AND NOT IN1;
27.
28. %MEND SAMPLE;
29.
30. DATA _NULL_; SET CASEOUT;
31. CALL EXECUTE
    ('%SAMPLE('||AGE||','||GENDER||','||COUNT||')');
32. RUN;
```

The selection of controls occurs in Lines 12-16 in Listing 4. Specifically, controls are selected from the now randomized QUALIFY1 subset, starting with the first observation through the desired number of controls. Should there not be enough controls to meet the desired amount, then all the controls are selected (Lines 12-14). And for auditing purposes, the variable 'TAG' is created and populated with either 'YES' or 'NO', depending on whether or not the number of matched controls was attained.

The resulting subset of matched controls, 'QUALIFY2', is appended to a data set comprised of other successful matches ('MATCHES') (Line 18). This file is then sorted by the controls unique identifying variable, as is the original CONTROLS file created in the second step of this program. The matched controls are then removed from the CONTROLS file in a merge statement (Lines 24-26). The macro ends on Line 28 in Listing 4.

This macro is conditionally executed, as the CALL EXECUTE feature passes the matching criteria singly though to the macro (Lines 30-32). Again, this matching criteria is comprised of a unique list of matching values, along with the corresponding frequencies that these combinations are present in the case population (refer to Step 3).

The fifth and final step of this program is conducted for auditing purposes. The objective here is to produce a report detailing instances in which the desired number of controls is not met. This is accomplished by merging the files containing unique list of values of the matching variables for the cases and for the matched controls. The list for the matched controls is generated in Lines 1-2 of Listing 5. A variable, 'DIFF', is created to represent the difference between the expected ('CON_NEED') and actual ('CON_CNT') number of matched controls (Line 15). This difference is then presented in a summary report (Lines 17-19). A sample of this report is presented Appendix C and it presents two different scenarios (a case-control ratio of 1:5, matching on age and gender):

1.) four cases (CASE_CNT) comprised of 40 year old (CASE_AGE) females (CASE_SEX) and matched with 15 controls (CON_CNT), 5 less (DIFF) than the 20 expected (CON_NEED); and

2.) two 55-year-old males and expecting 10 matches but achieving none.

Listing 5. Step 6 of the Case-Control Matching Program.

```
1.    PROC FREQ NOPRINT DATA=MATCHES;
2.      TABLES CASE_AGE*CASE_SEX/OUT=CON_OUT;
3.
4.    PROC SORT DATA = CASEOUT
        (RENAME=(AGE=CASE_AGE GENDER=CASE_SEX
5.      COUNT=CASE_CNT)); BY CASE_AGE CASE_SEX;
6.
7.    PROC SORT DATA = CON_OUT
        (RENAME=(COUNT=CON_CNT));
8.      BY CASE_AGE CASE_SEX;
9.
10.   DATA FINAL(DROP=PERCENT);
11.     MERGE CASEOUT(IN=IN1) CON_OUT(IN=IN2);
12.       BY CASE_AGE CASE_SEX; IF IN1;
13.
14.   CON_NEED = CASE_CNT*&RATIO;
15.   DIFF = CON_CNT-CON_NEED;
16.
17.   PROC PRINT DATA=FINAL;
18.     WHERE DIFF < 0;
19.       TITLE 'INSUFFICIENT MATCHES';
```

The CASES and MATCHES files can then be joined into a single file. This file is now ready for a matched analysis.

Discussion

The Case-Control Matching Program presented here was designed to match on age ranges, as we commonly do in our own investigations. To match on individual ages, simply enter '0' for AGERANGE value (Listing 1) or remove the AGERANGE reference in the WHERE statement (Listing 4). However, users are not obligated to use this criterion and can easily account for ranges of other factors (e.g., height, weight, etc.). The key steps to address this are the following: (1) establish a global macro variable(s) for this range(s) (Step 1) and (2) apply this macro variable(s) in the WHERE statement (Step 4).

The Case-Control Matching Program also incorporates the RANUNI function to randomize the controls, thus ensuring an unbiased selection process (Step 4). Specifically, controls who are qualified for matching are assigned a random number by this function and are then resorted according to this number. The objective here is to take an exact-sized random sample without replacing of successful matches. It should be noted that there are additional approaches to random sampling.[3] You are encouraged to explore these approaches should your selection objectives differ from ours.

The Case-Control Matching Program can also be modified to match each case one at a time, rather than combinations of values. This can be accomplished in various ways but most simply by including the UNIQUEID variable in the PROC FREQ step in Step 4 (Listing 4).

We did opt to match cases and controls based on unique combinations of values in the case population rather on a case-by-case basis to minimize resource requirements. To illustrate this, we ran this program for 2,958 cases and 63,258 controls, matching on age (±1 year) and gender at a case-control ratio of 1:5. We found the combination approach executed the macro (Step 4) 94.25% fewer times and resulted in a 75.04% reduction in the execution time (Table 1). Nevertheless, either methodology will produce the same results.

Table 1. Matching Methodologies Performance Comparison.

Method	Iterations	Execution Time (minutes)
Combination	170	19.97
Case-By-Case	2,958	80.00

Conclusion

We have introduced a SAS program that automates the matching of controls with cases. The Case-Control Matching Program overcomes several limitations of a previous SAS program by accommodating multiple matching variables and by varying the number of controls matched per case. The combination of these two features lowers the chances of confounding and increases the statistical power of the study, respectively.

Due to the popularity of matching cases and controls, we strongly recommend that investigators not overlook the importance of conducting the appropriate matched design and analysis. Nevertheless, we believe the Case-Control Matching Program is a powerful and flexible tool, which will help minimize the burdens of matching controls with cases.

References

1. Schlesselman JJ. (1982), *Case-Control Studies: Design, Conduct, Analysis*, New York: Oxford U.
2. Tassoni C, Chen B, and Chu C. (1997) “One-to-One Matching of Case/Controls Using SAS Software,” *Proceedings of the Twenty Second Annual SAS Users Group International Conference*, 22, Paper 257.
3. SAS Institute Inc. (1991), *SAS® Language and Procedures: Usage 2, Version 6, First Edition.* Cary, NC: SAS Institute Inc.

Contact Information

Edgar Mounib
25 Marion Road
Belmont, MA 02478
Edgar@Mounib.com
(617) 832-4976 (work)

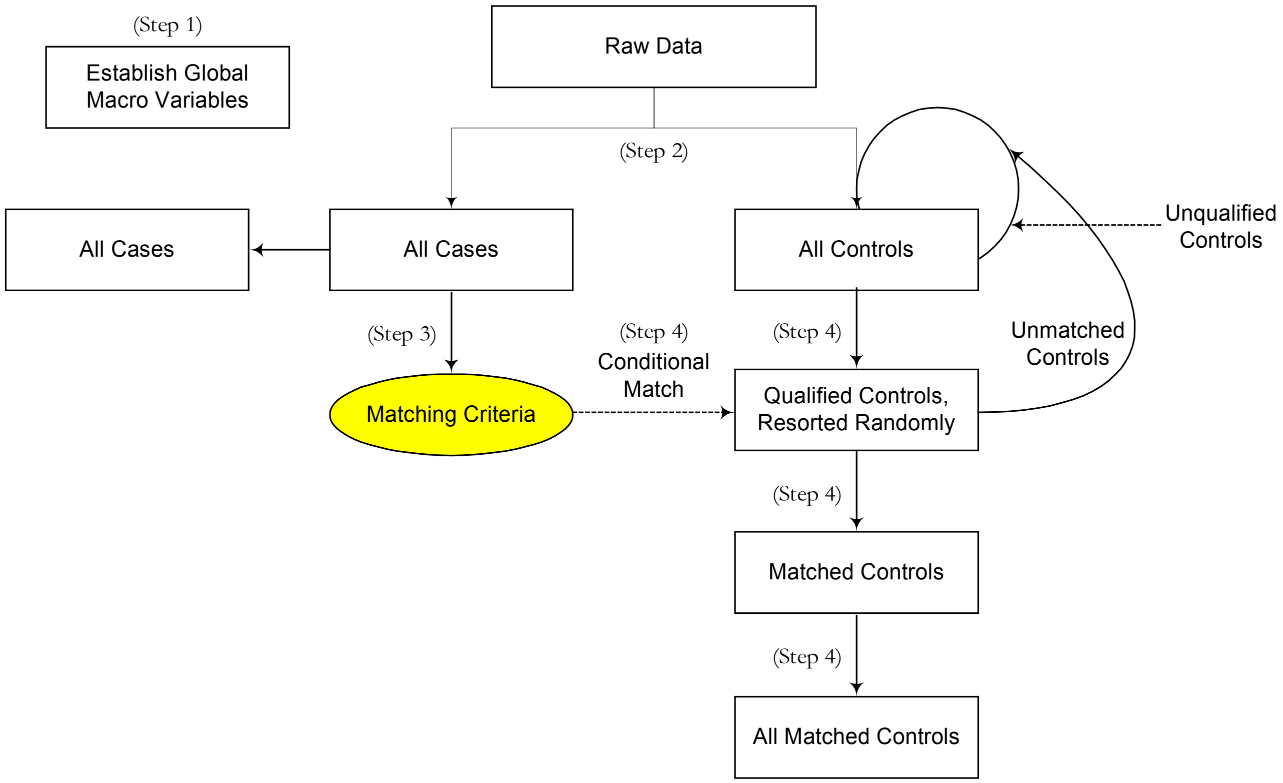
Thiru Satchi
150 Rumford Avenue
Mansfield, MA 02048
TSatchi@hotmail.com
(617) 832-3432 (work)

Appendix A

Case-Control Matching Program

Step	Line	Code
1	1.	%LET AGERANGE = 1;
	2.	%LET RATIO = 5;
2	1.	DATA CASES CONTROLS;
	2.	SET RAW_DATA;
	3.	
	4.	IF CASECTRL = 1 THEN OUTPUT CASES;
	5.	ELSE OUTPUT CONTROLS;
3	1.	PROC FREQ NOPRINT DATA=CASE;
	2.	TABLES **AGE*GENDER**/OUT=CASEOUT;
4	1.	%MACRO SAMPLE(**V_AGE,V_SEX**,V_COUNT);
	2.	DATA QUALIFY1; SET CONTROLS;
	3.	WHERE **(&V_AGE-&AGERANGE <=AGE<=&V_AGE+&AGERANGE)**
	4.	**AND (GENDER = "&V_SEX");**
	5.	
	6.	SEED=RANUNI(0);
	7.	PROC SORT; BY SEED;
	8.	
	9.	**CASE_AGE=&V_AGE;**
	10.	**CASE_SEX="&V_SEX";**
	11.	
	12.	DATA QUALIFY2;
	13.	SET QUALIFY1 NOBS=TOTOBS;
	14.	IF _N_ <= &V_COUNT*&RATIO;
	15.	IF &VCNT*&RATIO <= TOTOBS THEN TAG = 'YES';
	16.	ELSE TAG = 'NO';
	17.	
	18.	PROC APPEND BASE=MATCHES DATA=QUALIFY2;
	19.	
	20.	PROC SORT DATA=QUALIFY2 OUT=TEMP1 (KEEP= **UNIQUEID**); BY **UNIQUEID**;
	21.	
	22.	PROC SORT DATA=CONTROL OUT=TEMP2; BY **UNIQUEID**;
	23.	
	24.	DATA CONTROL;
	25.	MERGE TEMP1(IN=IN1) TEMP2(IN=IN2);
	26.	BY **UNIQUEID**; IF IN2 AND NOT IN1;
	27.	
	28.	%MEND SAMPLE;
	29.	
	30.	DATA _NULL_; SET CASEOUT;
	31.	CALL EXECUTE ('%SAMPLE('||**AGE||','||GENDER||','||COUNT**||')');
	32.	RUN;
5	1.	PROC FREQ NOPRINT DATA=MATCHES;
	2.	TABLES **CASE_AGE*CASE_SEX**/OUT=CON_OUT;
	3.	
	4.	PROC SORT DATA = CASEOUT (RENAME=(**AGE=CASE_AGE GENDER=CASE_SEX**
	5.	COUNT=CASE_CNT)); BY **CASE_AGE CASE_SEX**;
	6.	
	7.	PROC SORT DATA = CON_OUT (RENAME=(COUNT=CON_CNT));
	8.	BY **CASE_AGE CASE_SEX**;
	9.	
	10.	DATA FINAL(DROP=PERCENT);
	11.	MERGE CASEOUT(IN=IN1) CON_OUT(IN=IN2);
	12.	BY **CASE_AGE CASE_SEX**; IF IN1;
	13.	
	14.	CON_NEED = CASE_CNT*&RATIO;
	15.	DIFF = CON_CNT-CON_NEED;
	16.	
	17.	PROC PRINT DATA=FINAL;
	18.	WHERE DIFF < 0;
	19.	TITLE 'INSUFFICIENT MATCHES';

Appendix B
Schematic Diagram of the Case-Control Matching Program



(Step 1)
Establish Global Macro Variables

Raw Data

(Step 2)

All Cases

All Cases

All Controls

Unqualified Controls

(Step 3)

Matching Criteria

(Step 4)
Conditional Match

(Step 4)

Qualified Controls, Resorted Randomly

Unmatched Controls

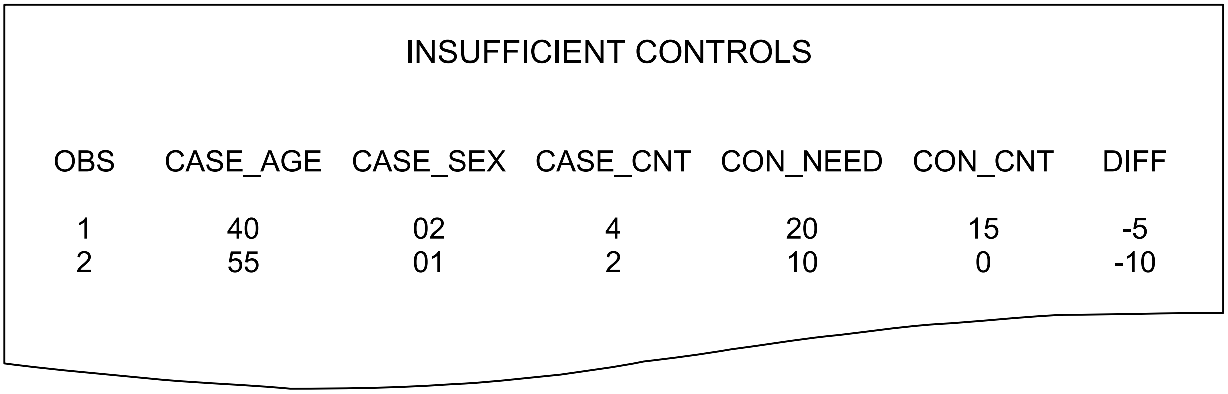
(Step 4)

Matched Controls

(Step 4)

All Matched Controls

Appendix C
Sample Output from the Step 5



```
                         INSUFFICIENT CONTROLS


OBS   CASE_AGE   CASE_SEX   CASE_CNT   CON_NEED   CON_CNT    DIFF

 1       40         02          4          20        15        -5
 2       55         01          2          10         0       -10
```