Using SAS to match cases for a case-control study

Wenbin Li
May 10, 2012
Outline

• Background Information
• Choosing Controls
• SAS Macros
Drug Safety and Effectiveness Network (DSEN)-1

DSEN has been established to increase the evidence on the post-market safety and effectiveness of drugs available to regulators, policy-makers, health care providers and patients.

- Canadian Institutes for Health Research
Drug Safety and Effectiveness Network (DSEN)-2

• Canadian Network for Observational Drug Effect Studies (CNODES)
• Prospective Studies
• Meta-Analysis
Effect of High Dose Statin Exposure on the Risk of Acute Kidney Injury

Nested Cohorts

1) Start of data availability
2) First possible date of entry into nested cohort (= start of data + 365 days)
3) Accrual of AKI outcome events begins
4) Last possible date of entry into nested cohort
5) Accrual of cases ends
Nested case control design

risk set: those who could have developed the disease, but did not (at the time when case occurred)

Madhukar Pai
Nested case control design

The study begins by selecting subjects based on

Exposed

Review records

Disease (cases)

Unexposed

Exposed

Review records

No disease (controls)

Unexposed

AFMC Primer on Population Health
Why Match?

• Improve efficiency of study
• Control for confounding
Matching Criteria

- For each case, identify all patients at risk
- Same sex, age within ± 2 years, cohort entry ± 90 days
- Widen the age criterion to obtain at least one match
How it work in SAS

CASES

Input datasets

1

CONTROLS

2

Later on …

3

Robert Matthews et al, 2004
Cases and Controls Datasets

*Cases and controls datasets;
proc sql;
    create table cases_cacol as
    select *, 1 as caco, monotonic() as match_num
    from dataset_case;
quit;

data _null_;  
    set cases_cacol end=last;  
    if last then call symput('N', match_num);
run;
%put &N;

data control_caco0;
    set dataset_control  
    caco=0;
run;
Cases and Controls Datasets-2

%macro first_run ;
  %do match_num =1 %to &N;

  data cases_cacol_&match_num;
    set cases_cacol;
    if match_num=&match_num;
  run;

  proc sql;
    create table nocase&match_num as
      select a.*
      from cases_cacol a,
        cases_cacol_&match_num b
      where a.event_date > b.event_date;
  quit;

  data control_nocase&match_num;
    set nocase&match_num (drop=event_date match_num);
    if caco=1 then caco=0;
  run;
Cases and Controls Datasets-3

data riskset_{&match_num};
set control_caco0
  control_nocase{&match_num};
run;

data riskset_random_{&match_num};
set riskset_{&match_num};
  rand_num=RANUNI(123);
run;
Matching-1

*matching;
proc sql;
  create table controls_match_num as
  select
    a.key_hsn as case_hsn,
    b.key_hsn as control_hsn,
  from cases_cacol_match_num a,
  riskset_random_match_num b
  where Matching Criteria

proc sort data=controls_id_match_num;
by case_hsn rand_num; run;

data controls_enough_match_num
  not_enough_match_num;
  set controls_match_num;
  by case_hsn;
  retain num;
  if first.case_hsn then num=1;
  if num le 10 then do;
    output controls_enough_match_num;
    num=num+1;
  end;
  if last.case_hsn then do;
    if num le 10 then output
    not_enough_match_num;
  end; run;
Matching-2

proc append base=temp.not_enough
data=not_enough_match_num; run;
proc append base=temp.aki730_match
data=controls_enough_match_num; run;
proc datasets library=work kill; run;
quit;
@end;
%mend;
%do_first;

*Collect all cases that do not qualify for the first run;
proc sql;
    create table temp.no_match_1st as
    select *
    from temp.nkd_aki730_cacol
    where match_num not in (select
      match_num from temp.nkd_aki730_match);
quit;

/*all cases got the matching controls*/

******* Second run *******
Conclusion

SAS macro can be simple but still could achieve desired results
References

Madhukar Pai, Case-control studies

Robert Matthews et al, 2004. SAS® Programs to Select Controls for Matched Case-Control Studies

Hugh Kawabata et al, . Using SASÒ to Match Cases for Case Control Studies
Questions?

Thank you!