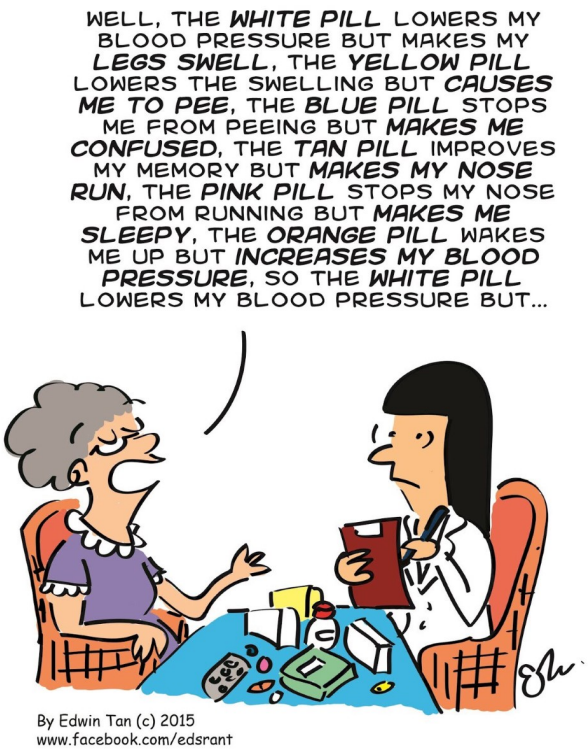
Prescription Sequence Symmetry Analysis SAS Macro

to Identify Potential Prescribing Cascades



Instruction Guide

Earl J. Morris, PharmD, MPH

Advisor: Scott M. Vouri, PharmD, PhD, BCGP

University of Florida College of Pharmacy
Department of Pharmaceutical Outcomes and Policy

Table of Contents

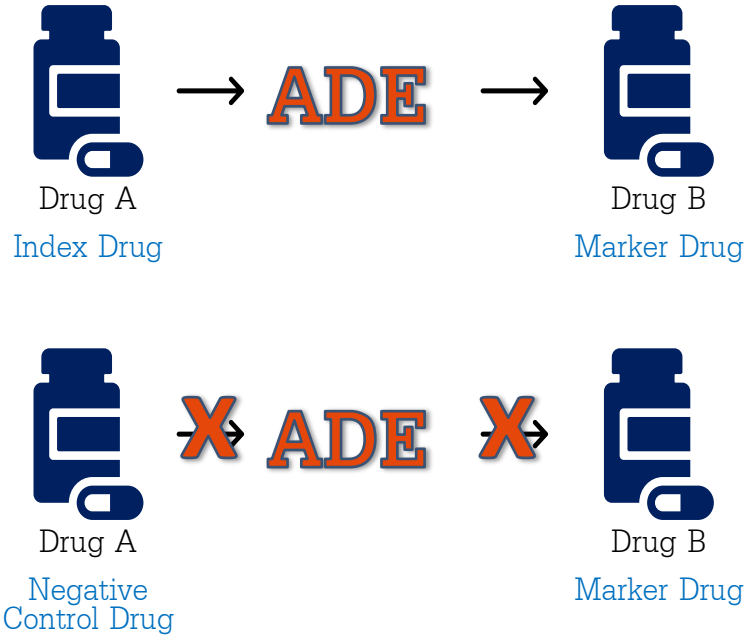
Overview

The PSSA method is a pharmacovigilance tool used to rapidly identify adverse event signals within large administrative health databases [1]. This method can be used to evaluate the association of a medication (index drug) to a related adverse event and subsequent prescription of a second drug (marker drug) and employs a case-only design including only patients who are users of both drugs of interest (i.e. index drug and marker [e.g., outcome] drug) [2, 3]. In assessing the association between the index drug and marker drug, PSSA uses a population of new users of both drugs and compares the number of subjects who used the index drug before the marker drug to the number of users who used the marker drug before the index drug. [1] Therefore, risk is estimated by calculating the ratio (adjusted sequence ratio [aSR]) of subjects who initiate index drug --> marker drug to subjects who initiate marker drug --> index drug, while accounting for secular trends of prescribing of these medications over time (i.e., adjusting for a null effect; Appendix S1).

We believe this macro is valuable to collaborators because it can be run in a relatively straightforward manner, is flexible, and can be used to identify adverse event signals and inappropriate prescribing in any clinical area.

The macro is written using SAS programming language and is optimized to analyze IBM MarketScan Research Databases, including commercial claims and Medicare Supplemental claims. The main output of this program is the adjusted sequence ratio(s) for hypothesized index drug --> marker drug associations.

Terminology



Note: When designing a PSSA analysis, incorporating negative control (e.g., absence of an adverse drug event [ADE] is well established) dyads in separate analyses may help assess for potential biases. In studies testing a specific hypothesis, controls should ideally be chosen by identifying negative control drug – marker drug dyads where the index drug replacement is used in a similar population with mutual indications and similar healthcare follow-up patterns as the study index drug [24, 74]. In particular, negative controls may be useful in identifying instances of time-varying bias, protopathic bias, and confounding by indication [12, 29, 50].

Customizing the %let statements

LOCATED AT THE VERY TOP OF THE CODE

Because the most efficient way to run this macro is to submit the customized full macro code in batch mode, the first part of this instruction guide will provide tips and instructions on how to customize the code. Before running the macro, there are a number of let statements that need to be customized based on the index drug --> marker drug dyad of interest.

Update your libname

Remember to update the libname and destination path for the folder you desire.

libname yourname '/data/resultdata/filedestinationpath/';

%let library

Input the libname for the file destination folder you chose above.

Example:

%let library = yourname;

%let redbook

Input the most current Redbook file name in the Truven folder.

Example:

%let redbook = redbook_18;

%let enrollment

Input the most current eligibility file name in the Truven folder.

Example:

%let enrollment = truven_cts_elig_rx_cc_mc_0519;

%let cc

Input Y to include IBM MarketScan Commercial Claims. Input N to exclude IBM MarketScan Commercial Claims.

Options:

- Y
- N

Example:

%let cc = Y;

%let mc

Input Y to include IBM MarketScan Medicare Supplemental Claims. Input N to exclude IBM MarketScan Medicare Supplemental Claims.

Options:

- Y
- N

Example:

%let mc = Y;

*Since PSSA utilizes a case-only design, it is okay to include commercial and Medicare Supplemental claims.

%let index_mstmds

Input the Master Form Descriptions to be included for the *INDEX DRUG*, in single quotes separated by commas. Of note, 'Capsule' will include any drugs with Master Form Description starting with 'Capsule', i.e. 'Capsule, Delayed Release', etc.

Example:
%let index_mstmds = (‘Capsule’, ‘Tablet’);

%let index_roacd

Input the Route of Administration Codes to be included for the *INDEX DRUG*, in single quotes separated by commas.

Example:
%let index_roacd = (‘PO’);

%let marker_mstmds

Input the Master Form Descriptions to be included for the *MARKER DRUG*, in single quotes separated by commas. Of note, 'Capsule' will include any drugs with Master Form Description starting with 'Capsule', i.e. 'Capsule, Delayed Release', etc.

Example:
%let marker_mstmds = (‘Capsule’, ‘Tablet’);

%let marker_roacd

Input the Route of Administration Codes to be included for the *MARKER DRUG*, in single quotes separated by commas.

Example:
%let marker_roacd = (‘PO’);

%let neg_control_mstmds

Input the Master Form Descriptions to be included for the *NEGATIVE CONTROL DRUG*, in single quotes separated by commas. Of note, 'Capsule' will include any drugs with Master Form Description starting with 'Capsule', i.e. 'Capsule, Delayed Release', etc.

Example:
%let neg_control_mstmds = (‘Capsule’, ‘Tablet’);

%let neg_control_roacd

Input the Route of Administration Codes to be included for the *NEGATIVE CONTROL DRUG*, in single quotes separated by commas.

Example:
%let neg_control_roacd = (‘PO’);

%let enrollment_before

Input the number of days of continuous enrollment BEFORE index drug initiation to be required.

Example:

%let enrollment_before = 360;

%let enrollment_after

Input the number of days of continuous enrollment AFTER index drug initiation to be required.

Example:

%let enrollment_after = 360;

%let diag_before

Input the number of days BEFORE index drug initiation to be searched for diagnosis codes.

Example:

%let diag_after = 360;

%let diag_after

Input the number of days AFTER index drug initiation to be searched for diagnosis codes.

Example:

%let diag_after = 360;

*diag_before and diag_after cannot be larger than enrollment_before and enrollment_after defined above.

%let exc_diag_yn

Input Y if excluding patients with a certain algorithm of ICD codes. Input N if NOT excluding patients with a certain algorithm of ICD codes.

Options:

- Y
- N

Example:

%let exc_diag_yn = Y;

%let exc_diag_icd

Input the array of ICD-9 and ICD-10 codes to be excluded in single quotes separated by commas. Of note, this will exclude patients with >/= 1 claim with a diagnosis code in the algorithm of interest.

If no exclusion criteria on diagnosis code, just enter ('1')

Example:

%let exc_diag_icd = (‘428’, ‘I501’);

%let comorbid_1_yn
%let comorbid_2_yn
%let comorbid_3_yn

Input Y if subgrouping patients with >/= 1 algorithm(s) of ICD codes.
Input N if NOT subgrouping patients with a certain algorithm of ICD codes.

Of note, this macro allows you to subgroup patients using up to three different algorithms of ICD-9 and ICD-10 codes.

If no diagnosis subgroups of interest, input...

```
% let comorbid_1_yn = N;
% let comorbid_2_yn = N;
% let comorbid_3_yn = N;
```

If only interested in one diagnosis subgroup, input...

```
% let comorbid_1_yn = Y;
% let comorbid_2_yn = N;
% let comorbid_3_yn = N;
```

If two subgroups of interest, input...

```
% let comorbid_1_yn = Y;
% let comorbid_2_yn = Y;
% let comorbid_3_yn = N;
```

If three subgroups of interest, input...

```
% let comorbid_1_yn = Y;
% let comorbid_2_yn = Y;
% let comorbid_3_yn = Y;
```

%let comorbid_icd_1
%let comorbid_icd_2
%let comorbid_icd_3

Input the array of ICD-9 and ICD-10 codes to be subgrouped in single quotes separated by commas. Of note, this will subgroup patients with >/= 1 claim with a diagnosis code in the algorithm of interest.

Therefore, if comorbid_i_yn = Y, then comorbid_icd_i should contain an array of ICD codes.
If comorbid_i_yn = N, then comorbid_icd_i should simply contain ('1'). (It really doesn't matter what is within the parentheses if comorbid_i_yn = N.)

Example: ('428', 'I501');

%let age_low

Input the minimum age in years an individual must be to be included in the analysis

Example:

%let age_low = 18;

%let age_high

Input the maximum age in years an individual must be to be included in the analysis

Example:

%let age_high = 90;

%let PSSA_before_after_1
%let PSSA_before_after_2
%let PSSA_before_after_3

Input the maximum number of days required between index drug and marker drug initiation to be included in the analysis. Of note, this macro allows you to conduct up to three different analyses with different time intervals. Jesper Hallas, who published the proof-of-concept paper for PSSA, recommends a max time interval of 360 days.

If conducting PSSA analyses with multiple time intervals, order them in descending order.

For example, if conducting three PSSA analyses with time intervals of 360 days, 180 days, and 90 days...

```
%let PSSA_before_after_1 = 360;
%let PSSA_before_after_2 = 180;
%let PSSA_before_after_3 = 90;
```

If conducting two PSSA analyses with time intervals of 360 days and 180 days...

```
%let PSSA_before_after_1 = 360;
%let PSSA_before_after_2 = 180;
%let PSSA_before_after_3 = 0;
```

If conducting one PSSA analysis with time interval of 180 days...

```
%let PSSA_before_after_1 = 180;
%let PSSA_before_after_2 = 0;
%let PSSA_before_after_3 = 0;
```

* PSSA_before_after_1 cannot be larger than enrollment_before and enrollment_after defined above.

%let CI

Input the confidence interval (number only [without %]) for adjusted sequence ratio desired.
To account for multiple testing, some authors use 99% confidence intervals.
This macro allows options of 90%, 95%, 99%, and 99.9%.

Example:

```
%let CI = 99;
```

%let year_1
%let year_2
%let year_3
%let year_4

Although PSSA inherently adjusts for secular trends in prescribing of drugs, it is helpful to investigate if there are any major differences in adjusted sequence ratios among subgroups of the total number of years of data included.
This macro conducts subgroup analyses among three subgroups of year ranges.

Let year_1 equal the first year of data included.
Let year_2 equal the lower limit of subgroup 2.
Let year_3 equal the lower limit of subgroup 3.
Let year_4 equal the upper limit of subgroup 3.

For example, the below let statements would make subgroup 1 be 2005-2010, subgroup 2 be 2011-2014, and subgroup 3 be 2015-2019.

```
%let year_1 = 2005;
%let year_2 = 2011;
%let year_3 = 2015;
%let year_4 = 2018;
```

%let min_days

%let max_days

To perform the PSSA analysis, SAS requires the stored value for the first and last date of data included in the study. In other words, this is the number of elapsed days since January 1, 1960. Let min_days equal the stored value for the first day of data included. Let max_days equal the stored value for the last day of data included.

Use https://www.sastipsbyhal.com/2012/01/sas-date-calculator-now-available.html and replace 'Date: 1/1/1960' with the date of interest to find these values.

Example:

```
%let min_days =16437; (January 1, 2005)
%let max_days =21914; (December 31, 2019)
```

%PSSA macro

LOCATED AT THE VERY BOTTOM OF THE CODE

```
%macro PSSA(yr_v,
            index_class,
            index_drugs,
            marker_class,
            marker_drugs,
            neg_control_class,
            neg_control_drugs,
            subgrp);
```

yr_v

Input the year and corresponding versions separated by spaces for included IBM MarketScan datasets.

Example:

05_4 06_3 07_3 08_3 09_3 10_3 11_3 12_2 13_2 14_1 15_1 16_1 17_1 18_1 19_1

index_class

The character value of this does not really matter. It is used to assign dataset names for the index drug class. Therefore, it should be short (</=3 characters) and unique from marker_class and neg_control_class.

Example:

If the index drug class of interest includes all calcium channel blockers, input...
CCB

If the index drug class of interest includes amiodarone alone, input...
AMI

index_drugs

This is a vector of the generic names in all capital letters of index drugs included in the analysis separated by a space. If there is only one index drug, simply enter the generic name of that drug in all capital letters.

Example:

AMLODIPINE FELODIPINE NICARDIPINE (multiple)

or

AMIODARONE (single)

marker_class

The character value of this does not really matter. It is used to assign dataset names for the marker drug class. Therefore, it should be short (</=3 characters) and unique from index_class and neg_control_class.

Example:

If the marker drug class of interest includes all loop diuretics, input...
LOO

If the marker drug class of interest includes thyroxine alone, input...
THY

marker_drugs

This is a vector of the generic names in all capital letters of marker drugs included in the analysis separated by a space. If there is only one marker drug, simply enter the generic name of that drug in all capital letters.

Example:

FUROSEMIDE BUMETANIDE (multiple)

or

THYROXINE (single)

neg_control_class

The character value of this does not really matter. It is used to assign dataset names for the negative control drug class. Therefore, it should be short (</=3 characters) and unique from index_class and marker_class.

Example:

If the negative control drug class of interest includes all ACE inhibitors, input...
ACE

If the negative control drug class of interest includes rosuvastatin alone, input...
ROS

neg_control_drugs

This is a vector of the generic names in all capital letters of negative control drugs included in the analysis separated by a space. If there is only one negative control drug, simply enter the generic name of that drug in all capital letters.

Example:

LISINOPRIL FOSINOPRIL ENALAPRIL (multiple)

or

ROSUVASTATIN (single)

subgrp

Input the subgroups of interest for which you want the sequence ratios reported separated by spaces.

Options:

- main (**must include**)
- 1849 (individuals 18-49 years old)
- 5064 (individuals 50-64 years old)
- lt65 (individuals < 65 years old)
- ge65 (individuals >/= 65 years old)
- male
- female
- yr_1 (range of years - subgroup 1)
- yr_2 (range of years - subgroup 2)
- yr_3 (range of years - subgroup 3)
- com1 (comorbidity subgroup 1; *Only include this if %let comorbid_1_yn = Y.*)
- no_com1 (those without comorbidity 1; *Only include this if %let comorbid_1_yn = Y.*)
- com2 (comorbidity subgroup 2; *Only include this if %let comorbid_2_yn = Y.*)
- no_com2 (those without comorbidity 2; *Only include this if %let comorbid_2_yn = Y.*)
- com3 (comorbidity subgroup 3; *Only include this if %let comorbid_3_yn = Y.*)
- no_com1 (those without comorbidity 3; *Only include this if %let comorbid_3_yn = Y.*)

Example:

main 1849 5064 lt65 ge65 male female yr_1 yr_2 yr_3 com1 no_com1 com2 no_com2 com3 no_com3

or

main 1849 5064 lt65 ge65 male female yr_1 yr_2 yr_3

Following the two examples above, the %PSSA macro would look like this:

```
%PSSA(05_4 06_3 07_3 08_3 09_3 10_3 11_3 12_2 13_2 14_1 15_1 16_1 17_1 18_1 19_1,
      CCB,
      AMLODIPINE FELODIPINE NICARDIPINE,
      LOO,
      FUROSEMIDE BUMETANIDE,
      ACE,
      LISINOPRIL FOSINOPRIL ENALAPRIL,
      main 1849 5064 lt65 ge65 male female yr_1 yr_2 yr_3);
```

```
%PSSA(05_4 06_3 07_3 08_3 09_3 10_3 11_3 12_2 13_2 14_1 15_1 16_1 17_1 18_1 19_1,
      AMI,
      AMIODARONE,
      THY,
      THYROXINE,
      ROS,
      ROSUVASTATIN,
      main 1849 5064 lt65 ge65 male female yr_1 yr_2 yr_3);
```

Initiating the PSSA Macro

Using the Macro:

The most efficient way to use the PSSA macro is to customize the macro and run the macro in batch mode.

Update all of the %let statements to the parameters you desire and update the %PSSA macro to identify the drugs of interest. Then save the code and run it in batch mode.

The code will output a .lst file containing:

- the number of individuals who initiated index drug --> marker drug
- the number of individuals who initiated marker drug --> index drug
- crude sequence ratio (cSR)
- null effect sequence ratio (nSR)
- adjusted sequence ratio (aSR)
- lower limit of adjusted sequence ratio (aSR_ll)
- upper limit of adjusted sequence ratio (aSR_ul)

  ...for every index drug/marker drug and negative control drug/marker drug analysis and subgroup analysis. Informative titles are provided to distinguish the subgroup analyses and corresponding sequence ratios.

References

1. Hallas J. Evidence of depression provoked by cardiovascular medication: a prescription sequence symmetry analysis. Epidemiology. 1996:478-84.

2. Hallas J, Pottegård A. Use of self‐controlled designs in pharmacoepidemiology. Journal of internal medicine. 2014;275(6):581-9.

3. Cadarette SM, Maclure M, Delaney JAC, Whitaker HJ, Wang SV, Hayes KN, et al. Control yourself: ISPE-sponsored guidance in the application of self-controlled study designs in pharmacoepidemiology. Pharmacoepi Drug Saf. 2020.

Appendix S1: PSSA Equations

1) Crude Sequence Ratio (cSR)

$$cSR = \frac{n_{index \rightarrow marker}}{n_{marker \rightarrow index}}$$

where:

$n_{index \rightarrow marker}$ = number of individuals prescribed index drug before marker drug, and

$n_{marker \rightarrow indexr}$ = number of individuals prescribed marker drug before index drug.

2) Overall Average Probability of Marker Drug Prescribed after Index Drug, Given Background Prescribing Trends (P_a)

$$P_a = \frac{\sum_{m=1}^{u}[I_m \times (\sum_{n=m+1}^{m+d} M_n)]}{\sum_{m=1}^{u}[I_m \times (\sum_{n=m-d}^{m-1} M_n + \sum_{n=m+1}^{m+d} M_n)]}$$

where:
m = a given day within the study period (exposure window)
u = last day of study period
I_m = number of incident users of index drug on a given day
d = specified number of days within the study period (exposure window)
n = consecutive days of the study period (exposure window)
M_n = number of individuals receiving first marker drug on a given day

3) Null-effect Sequence Ratio (neSR)

$$neSR = \frac{P_a}{(1 - P_a)}$$

where:
P_a = overall average probability of marker drug prescribed after index drug, given background prescribing trends (calculated above)

4) Adjusted Sequence Ratio (aSR)

$$aSR = \frac{cSR}{neSR}$$

where:
cSR = crude sequence ratio (calculated above)
nesR = null-effect sequence ratio (calculated above)