PH-92: SAS Programming And Generic Techniques for Cohort Creation and Consort Diagrams

Brandy is a statistician and database programmer at the University of Michigan Medical School. She has worked on a wide variety of research projects. She holds a bachelors degree in engineering, and masters degrees in mathematics and statistics. The American Public Health Association awarded Brandy a student research award for her statistics masters thesis on path analysis. She has also received "Advanced Analytics" and "Data for Good" awards from the Midwest SAS Users Group conference.

Outside of work, she enjoys playing the piano, composing music, singing in choir, yoga, tai chi, and martial arts aerobics.

Outline

1)_Consort Diagram.

2)_Two Data Steps. Compute Variables, Followed by Selection and Deletion

3)_Useful Programming Techniques for Computing Variables.

Attribute Statements

Naming Variables After The Creators of the Coding Algorithm

Converting Character Variables to Numeric & Time Variables Starting at Zero.

4)_Selection of Cohort.

Where Statement

SAS Format for Diseases or Procedures.

Average Cases Per Facility.

5)_Closing Comments.

Consort Diagram

- Inclusion and Exclusion Criteria for a Study or Analysis Project.
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4520133/#:~:text=CONSORT%20flow%20diagram%20of%20the,up%2C%20and%20data%20analysis).

All Patients in Breast National Cancer Database (NCDB)
2012 – 2020 (N = 2,180,804)

Inclusion criteria

Criterion 1: Female (N = 2,162,479)
Criterion 2: Age ≥ 18 (N = 2,162,479)
Criterion 3: Care at same facility (N = 2,066,281)
Criterion 4: 1–40 Lymph Nodes Examined (N = 1,553,756)
Criterion 5: Clinical N1 – N2,disease (N = 192,567)
Criterion 6: Axillary surgery (N = 184,552)
Criterion 7: Neoadjuvant chemotherapy (N = 80,612)
Criterion 8: Pathologic T0-T4 disease (N = 73,151)
Criterion 9: Clinical T1-T4 disease (N = 71,979)
Criterion 10: Histology invasive ductal, lobular (N = 71,932)

(N = 71,977)

Exclusion Criteria

Criterion 1: Clinical stage 4 (N = 70,997)
Criterion 2: Inflammatory breast cancer (N = 67,383)
Criterion 3: Clinical T0 disease (N = 67,383)
Criterion 4: Facility avg annual cases<10 (N = 67,365)

(N = 4,612)

All patients meeting selection criteria
(N = 67,365)

Slide 3

Compute Variables, Then Select and Delete

/* Step 1: SAS Data Step to Compute Variables. */

Proc Format;

Value TW2 1=“Ductal” 2=“Lobular” 3=“Both Ductal and Lobular” 9=“Other”; Run;

- data NCDB_PUF0_CompVar;
- set library.NCDB_PUF;

/* Attributes enable assigning label and format in a single statement */

- Attrib AgeDiag Label="Age at Diagnosis" Format=7.0;
- Attrib Grade_TW2 Label='Tumor Grade, Ton Wang Method' Format=TW2.;
- Attrib YearsAfter2012 Label="Years After 2012" Format=7.0;
- Attrib YearsAfter2012_2 Label="Years After 2012 sqr" Format=7.0;
- Attrib YearsAfter2012_3 Label="Years After 2012 cube" Format=7.0;
- Attrib Yr Label='Year of Diagnosis' format=7.0;

Useful Programming Techniques (1 of 2)

Name Variable After the Creator of the Algorithm with Their Initials.

Attrib Grade_TW2 Label='Tumor Grade, Ton Wang Method' Format=TW2.;

When coding is not intuitively obvious, helps to have a record of who created the coding scheme.

- /* Compute Histology_TW2 */
- if HISTOLOGY in ('8022', '8035', '8230', '8500', '8501', '8502', '8503', '8504', '8507', '8508', '8523') then Histology_TW2 =1; /* ductal */
- if HISTOLOGY in ('8520', '8521', '8524', '8525') then Histology_TW2=2; /* lobular. */
- if HISTOLOGY = '8522' then Histology_TW2=3; /* both. */
-
- if HISTOLOGY not in ('8022', '8035', '8230', '8500', '8501', '8502', '8503', '8504', '8507', '8508', '8520', '8521', '8522', '8523', '8524', '8525') then Histology_TW2=9; /* Others */

Useful Programming Techniques (2 of 2)

/* Convert Year of Diagnosis from Character to Numeric Variables */

Yr = YEAR_OF_DIAGNOSIS + 0;

/* Compute time variable to start at zero. */

/* Starting time variable at zero can help with model convergence, especially in longitudinal models. */

YearsAfter2012 = yr - 2012;

YearsAfter2012_2 = YearsAfter2012**2;

YearsAfter2012_3 = YearsAfter2012**3;

Can also compute age to start at 0, such as YearsOver18 = Age – 18.

Where Statement in SAS Data Step

- In a SAS data step, the “Where” statement executes faster than the “If” statement.
- Reference: https://support.sas.com/resources/papers/proceedings/proceedings/sugi31/238-31.pdf

- After building the dataset with computed variables, apply the selection criteria.

```
Data NCDB_PUF0;
Set NCDB_PUF0_CompVar;
Where (2012 <= Yr <=2 020)
      AND SEX='2'  /* Female, Criterion 1 */
      AND (18<=AGE<=90)   /* Age 18+, Criterion 2 */
      AND not (CLASS_OF_CASE='00') /* Care at same facility, Criterion 3 */
      AND ('01'<=REGIONAL_NODES_EXAMINED<='40');  /* Criterion 4 */
Run;
```

Use of Where, Put, SAS Format to Select Diseases, 1 of 4

/* View of Format File*/

fmtname	start	end	label
$app_icd10dx	K352	K352	appendicitis
$app_icd10dx	K353	K353	appendicitis
$cho_icd10dx	K8000	K8001	cholecystitis
$cho_icd10dx	K8012	K8013	cholecystitis
$cho_icd10dx	K8042	K8043	cholecystitis
$div_icd10dx	K5720	K5721	diverticulitis
$div_icd10dx	K5732	K5733	diverticulitis

Use of Where, Put, SAS Format to Select Diseases, 2 of 4

/* Import Format File Into SAS */

- Either use the file import wizard by clicking on file / import or use Proc Import.

- PROC IMPORT OUT= WORK.ICD10 DATAFILE= “H:\CHOP\CHOPAnalystPlaybook\BuildCohort_SAS_Generic\EGS_ICD10.xlsx"
- DBMS=EXCEL REPLACE; /* Tell SAS to import from Excel & replace prev file */
- RANGE="ICD10"; /* Using a range or guessingrows= is helpful */
- GETNAMES=YES; /* Get variable names from 1st row. */
- RUN;

*** Convert Dx10 to SAS formats ***;

proc format library=work cntlin=ICD10 (keep=fmtname start end label) ; run;

Use of Where, Put, SAS Format to Select Diseases, 3 of 4

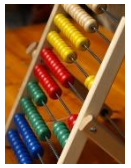
***** Convert Dx10 to SAS formats, fmtlib option with display the format ***;**

proc format library=work cntlin=ICD10 (keep=fmtname start end label) fmtlib; run;

START	END	LABEL
K3520	K3521	appendicitis
K353	K353	appendicitis
K3530	K3530	appendicitis
K651	K651	cholecystitis
K653	K653	cholecystitis
K659	K659	cholecystitis
K5720	K5721	diverticulitis
K5732	K5733	diverticulitis

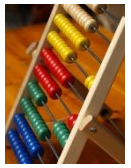
Use of Where, Put, SAS Format to Select Diseases, 4 of 4

```
*** Keep patients who have 1+ of the specified diseases ***;
/* Initializations */
kp0=0;  /* Keep indicator */
appendicitis = 0;  cholecystitis = 0;  diverticulitis = 0;

array arrICD{25} DGNS_CD01-DGNS_CD25; /* ICD codes */

do i=1 to 25;
  if (put(arrICD[i], $APP_ICD10DX.) = 'appendicitis') appendicitis = 1;
  if (put(arrICD[i], $CHO_ICD10DX.) = 'cholecystitis') then  cholecystitis = 1;
  if (put(arrICD[i], $DIV_ICD10DX.) = 'diverticulitis') then diverticulitis = 1;
end;
If (appendicitis = 1 or  cholecystitis = 1 or diverticulitis = 1) then kp0 = 1;
```



Average Cases Per Facility 1 of 3

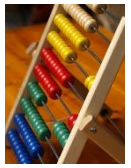
- **/* Compute Annual Hospital Volume */**
- Proc SQL;
- Create Table HospitalVolumeYear2012_2020 AS
- SELECT DISTINCT PUF_FACILITY_ID, YEAR_OF_DIAGNOSIS,
- Count(PUF_CASE_ID) as CaseCount label="Case Count" format=7.2
- FROM library.NCDB_PUF
- GROUP BY PUF_FACILITY_ID, YEAR_OF_DIAGNOSIS
- ORDER BY PUF_FACILITY_ID, YEAR_OF_DIAGNOSIS;
- Quit;



Average Cases Per Facility 2 of 3

- **/* Compute Average Hospital Volume by Year for New Data */**
- Proc SQL;
- Create Table HospitalAveVolume2012_2020 AS
- SELECT DISTINCT PUF_FACILITY_ID, Mean(CaseCount) as MeanCaseCount label="Mean Case Count" format=7.2
- FROM HospitalVolumeYear2012_2020
- GROUP BY PUF_FACILITY_ID
- ORDER BY PUF_FACILITY_ID; Quit;

- Data HospitalAveVolume2012_2020;
- set HospitalAveVolume2012_2020;
- if MeanCaseCount<10 then delete; Run;



Average Cases Per Facility 3 of 3

- **/* Exclude Hospitals with <10 average cases per year */**
- Proc Sort Data = NCDB_PUF0;
- By PUF_FACILITY_ID; Run; /* Begin with sort by facility id */

- Data NCDB_PUF_N1N2;
- Merge NCDB_PUF0(IN=N) library.HospitalAveVolume2012_2020(IN=C);
- by PUF_FACILITY_ID;

- **if (C=0 or N=0) then delete;**
- /* Delete if facility not in list where average volume > 10 */
- Run;

Closing Comments

- Attribute statement allows format and label in single statement.
- Naming variables after the person, who invented the coding system, can be helpful to remember who created the coding system.
- Using the “Where” statement in the data step has a faster execution time than the “If” statement.
- The use of a SAS format + put statement can shorten the SAS syntax to select a disease cohort.
- Minimum average annual case criteria can be programmed with Proc SQL, followed by a data step merge with the (IN=) option.

References

- CONSORT: when and how to use it. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4520133/#:~:text=CONSORT%20flow%20diagram%20of%20the,up%2C%20and%20data%20analysis)

- Faster data step execution time for “Where” compared to “If”. https://support.sas.com/resources/papers/proceedings/proceedings/sugi31/238-31.pdf

Contact Information

- Brandy R. Sinco, Statistician and Programmer/Analyst
- Michigan Medicine
- CHOP (Center for Healthcare Outcomes and Policy)
- Ann Arbor, MI 48109-2800
- E-Mail: brsinco@umich.edu