

The Anatomy of an Epidemiology Study

Paul Murray
FTP Software Consultants Ltd.
UK

Introduction

- Epidemiology/Health Outcomes
- Benefits of Studies
- Skills Required
- Data Sources
- A study

Epidemiology/Health Outcomes

➤ Health Outcomes

➤ Health Outcomes

- study of the end results of health services

➤ Epidemiology

- Studies the Distribution of Health Outcomes
- Patterns causes and effects of Health and disease conditions in populations

Benefits of Studies

- Information on real world use and practice
- Detect signals about benefits and risks of practices
- Formulate hypotheses for future experiments
- Provide part of data to design clinical trials
- Inform Clinical Practice

Skills

- Handling Large Data
- Summarise and Transform
 - SAS procedures
 - Data step manipulation
- Macro programming
- Statistics
- Report writing

Data Sources and Providers

Sources

- Claims Data
- Medical Data
 - GP
 - Hospital data

Providers

- IMS
- Mediplus
- Pharmetrics
- Thompson Reuters/Truven Health
- GPRD/CPRD
- Cegedim

Structure of Source Data

- Large datasets, hundreds of millions of records
- Tables per medical function
 - Enrollment
 - In-patient
 - Out-patient
 - Medication
- Heavily Coded
 - NDC codes
 - ICD codes
 - Readcodes
- Divided per year
- Multiple items of interest per record
 - Diag1-Diag10, drug1-drug5, enrol1-enrol12

A Study

- Background to study/study design
- Data source
- Cohort definition –
 - Inclusion/Exclusion Criteria
 - diagnosis
 - co-morbidity
 - treatment
 - procedures
 - drugs
 - study period
 - index date definition
 - pre and post index periods
 - demographics
 - Age on index date
- Control patients/matching
- Co-variates
- Statistical analysis
- Outputs – tables listings and figures
- Validation procedures

Using the Data

Data left “As-Is” to “Free Range” programmers

- projects can overlap in scope
- extraction can be slow - extract once
- multiple private marts build up taking up storage
- duplication of effort
 - programming
 - reporting
 - validation

Approach

- Vendor data can be transformed
- Transformations should be
 - Efficient
 - Understandable
 - Consistent
 - Easily validated
 - Additional structure

Marts

- Easy access to frequently needed data
- Creates collective view accessed by a group of users
- Improves end-user response time
- Ease of creation
- Lower cost than implementing a full data warehouse
- Potential users are more clearly defined than in a full data warehouse
- Contains only essential data, less cluttered.

Diagnosis

- Studies diagnosis driven
- Finite number of therapeutic areas
- Diagnosis defined by:
 - Single ICD
 - Multiple ICD Code
 - Combination
 - Diagnosis
 - Medical Procedure
 - Prescription

Use Meta-data (1)

Using SASHELP.views to find input table names

```
proc sql noprint;
/* number of tables */
  select trim(left(put(count(memname),best3.))) into : numtabs
  from sashelp.vtable
  where libname="DBASE"
    and substr(memname,1,4) = "HLTH" and substr(memname,5,1) in ("I","O");
/* names of tables */
  select memname into : tab1-:tab&numtabs
  from sashelp.vtable
  where libname="DBASE"
    and substr(memname,1,4) = "HLTH" and substr(memname,5,1) in ("I","O");
quit;
...
proc sql;
%do ix=1 %to &numtabs;
  create table ext&ix as
  select * from &&tab&ix
  where ICD10CODE in (select code from codelist);
quit;
%end;
```

Use Meta-data (2)

For the case of multiple diagnosis variables dx1-dxn

```
%do ix=1 %to &numtabs;
proc sql noprint;
/* find number of dx variables */
  select trim(left(put(count(*),best3.))) into : numdx
  from sashelp.vcolumn
  where libname="TR"
        and memname="&&tab&ix"
        and substr(name,1,2) ="DIAG";
quit;
data xd.&dsetout(label="01_DIAG_Patients &sysdate &systime" keep=patid enrolid year
date) ;
  set &dsetin(keep=patid enrolid year &datevar dx: rename=(&datevar=date)) ;
  if diag1="&diagcode" then output;
  %do iy=2 %to &numdx;
  else if diag&iy="&diagcode" then output;
  %end;
  drop i ;
run ;
%end;
```

Medication

- Medication driven study – medication marts
- Diagnosis driven study – faster access to medication data required
 - Partition by year
 - Split extract into smaller jobs
 - Indexes
- Indexes
 - MSGLEVEL=I
 - Message in log if index used
 - Suggestions about influencing use of index
 - Sort data in index order
 - Use of data set options to force index use
- IDXNAME=*index-name*
- IDXWHERE=YES

Enrolment/Eligibility

Included Patients

- Fully enrolled
- Regular events in database

➤ Sourced from across database and summarised

- Summary can be stored as but to save space

[illegible]

Enrolment Summary

creating a 12 bit string for each year

```
proc sql;
  create table xd.&dsetout._m(label="05_DIAG &sysdate &systime") as
  select patid,dobyr,sex,
  input(%do ix=1 %to 12;
    trim(left(put(enrol&ix,1.)))%if &ix ne 12 %then %do;!!%end;%end;,
    $binary16.) as enrolment length=2 format=$binary12.
  from &dsetin;
quit;
```

concatenating yearly to span the study

...

```
enrolment=input(%do ix=1 %to &numyears;
  put(enrolment&ix,$binary12.)
  %if &ix ne &numyears %then %do; !! %end;
  %end;
  ,$binary%eval(&numyears*12).);
```

For using the bit string to find a two year fully enrolled stretch

```
if index(put(enrolment,$binary60.),repeat("1",23));
```

Cohort/Controls

- Create a Cohort Mart
 - If extraction slow
 - If database updating

- Identify Controls from a large number of patients
 - Random sampling
 - Pair matching
 - Frequency/category matching

- Specified in SAP
 - Demographic data
 - Clinical indexes
 - PSM

Cost/Burden

- Actual financial cost
 - Inpatient
 - Outpatient
 - Medication
 - Hospitalisation
- Resource Usage
 - GP
 - Specialist
 - Diagnostic tests
 - ER
- Pill Burden
 - By drug group
 - Study related medication/Non Study Related

Summarise/Transpose

Summarise costs over any grouping required:

```
proc summary data=nz_cost nway;  
  class &patid &grps;  
  var cost;  
  format &grps &grpfmt;  
  where &daterange;  
  output out=&prepost._cost_&nmsuff(drop=_type_ _freq_) sum=&prepost._cost;  
run;
```

transpose to get 1 row per patient:

```
proc transpose data=&prepost._cost_&nmsuff out= tran_&prepost._cost_&nmsuff  
              prefix=&prepost._;  
  by &patid ;  
  id &grps;  
  format &grps &grpfmt;  
  var &prepost._cost;  
run;
```

Results

Example of output:

Obs	FATID	pre_cost	pre_I	pre_O	pre_D	pre_S	office	othout	pathrad	pre_en
1	27181101	11516.98	11399.05	117.92	.	.	117.92	.	.	.
2	28626706	5219.07	5184.03	32.15	2.88	.	.	32.151	.	.
3	29669001	8658.38	8632.10	16.07	10.20	.	16.07	.	.	.
4	48483301	3032.81	3032.81	.	.	.	.	.	.	.
5	48657002	33560.13	33411.63	148.50	.	.	148.50	.	.	124.88
6	49513201	8648.53	8323.23	.	325.29	.	.	.	220.45	.
7	53163101	11093.39	11024.44	.	68.94	.	.	.	.	.
8	53988801	11502.81	11484.41	12.08	6.32	.	12.08	.	.	76.12
9	55753702	6734.48	6623.11	63.22	48.14	.	.	63.221	.	.
10	113361201	28786.12	28783.58	.	2.54	.	.	.	.	.
11	156011202	2156.53	2080.15	76.37	.	.	76.37	.	.	98.75
12	156642802	27915.94	27826.00	89.94	.	.	.	89.941	.	.
13	160638802	13435.92	13401.71	3.24	30.96	.	.	3.245	.	.
14	186490601	4988.75	4988.75	.	.	.	.	.	12.34	.
15	190577501	7324.73	7094.66	.	230.06	.	.	.	.	.

Reporting

➤ TFLs defined in SAP

- Inclusion breakdown
- Study vs Controls
 - Demographics
 - Pre-post index
 - Costs
 - Pill burden
 - Co-morbidity
 - Drug class
 - Tests
 - Procedures
 - Resource use

A Report

Age group (years)				
50-59	6 (2.9%)	3 (2.8%)	3 (3.0%)	
60-69	37 (18.0%)	20 (18.9%)	17 (17.0%)	
70-79	89 (43.2%)	45 (42.5%)	44 (44.0%)	
80-89	64 (31.1%)	34 (32.1%)	30 (30.0%)	
90 and more	10 (4.9%)	4 (3.8%)	6 (6.0%)	
Mean (SD)	76.0 (8.6)	75.7 (8.7)	76.4 (8.5)	0.5265
Median (IQR)	76.0 (12.0)	76.0 (11.0)	76.5 (12.0)	0.5813
Gender				
Men	118 (57.3%)	61 (57.5%)	57 (57.0%)	1.0000
Women	88 (42.7%)	45 (42.5%)	43 (43.0%)	
Hometype				
1 Hospital	0 (0.0%)	0 (0.0%)	0 (0.0%)	
2 PNH	2 (1.0%)	1 (0.9%)	1 (1.0%)	
3 PRH	12 (5.8%)	4 (3.8%)	8 (8.0%)	
4 SRH/OPH	0 (0.0%)	0 (0.0%)	0 (0.0%)	
5 Vol. res home	0 (0.0%)	0 (0.0%)	0 (0.0%)	
6 Sheltered housing	0 (0.0%)	0 (0.0%)	0 (0.0%)	
7 Ind. living alone	41 (19.9%)	23 (21.7%)	18 (18.0%)	
8 Ind. living partner/spouse	136 (66.0%)	69 (65.1%)	67 (67.0%)	
9 Ind. living family	15 (7.3%)	9 (8.5%)	6 (6.0%)	
10 Independent/other	0 (0.0%)	0 (0.0%)	0 (0.0%)	
Status				
1 Single	30 (14.6%)	17 (16.0%)	13 (13.0%)	
2 Married	138 (67.0%)	71 (67.0%)	67 (67.0%)	
3 Divorced	4 (1.9%)	2 (1.9%)	2 (2.0%)	
4 Widowed	34 (16.5%)	16 (15.1%)	18 (18.0%)	
5 Separated	0 (0.0%)	0 (0.0%)	0 (0.0%)	

Reporting Macros

➤ Summary Macros

- Frequency counts
- Frequency tables
- Descriptive statistics
- Check class levels
- P values

➤ Formatting and output

- Frequency results and p values
- Statistical results and p values
- Spacing
- Titles
- Report output

Summary Macro

macro to get basic frequency counts for inclusion report, optional append to named result file:

```
%macro getcounts (order=,datasource=,label=,output=) ;  
proc sql;  
    create table row_&order as  
    select &order as order,"&label" as label length=120,count(distinct patid) as count  
    from &datasource;  
quit;  
%if &output ne %then %do;  
    proc append base=&output data=row_&order;  
    run;  
%end;  
%mend getcounts;
```

Statistical Macro

Part of a macro to get P-Values:

```
proc freq data =&data;
%if %upcase(&exact) eq EXACT %then %do;
    tables &class*&var / chisq fisher;
    ods output FishersExact=&out(rename=(nvalue1=prob) where=(index(label1,"Pr <=
P")>0));
%end;
%else %do;
    tables &class*&var / chisq;
    ods output chisq=&out( where=(statistic="Chi-Square"));
%end;
%if &format ne %then %do;
    format &var &format;
%end;
run;
```

Statistical Analysis

Example from a recent study:

- SAP defined:
 - Ancova analysis
 - Test multicollinearity
 - Homogeneity of Variance assumption
 - Homogeneity of regression slopes assumption
- If the tests fail a follow up analysis specified

Validation

- After thorough testing, independent validation.
 - Reported output represents original data
 - Reports match SAP specification
 - Logs with no errors or warnings
 - Test scripts and test data
 - Double programming
 - Code review
 - Output review

Macros

➤ Utility Macros

- Current output datasets
- Run times
- Delete tables
- Append tables

➤ Meta-data Macros

- Data set exists
- Variable exists
- Variable labels
- Variable types
- Variable formats

Utility Macros (1)

```
%macro dsetlist(lib=,nlabel=);  
proc sql;  
    create table sasuser.dsets_&sysdate as  
    select *  
    from sashelp.vtable  
    where libname="%upcase(&lib)" and memlabel is &nlabel  
    order by crdate descending  
    ;  
quit;  
%mend;  
  
%dsetlist(lib=xd,nlabel=null);  
%dsetlist(lib=xd,nlabel=not null);
```

Utility Macros (2)

```
infile folder(&prog..log) firstobs=%eval(&records-6) obs=%eval(&records-2)
      truncver recfm=v lrecl=256;
retain prog "&prog";
input line $256.;
template="00:00:00";
if index(line,"real time") then do;
    dotpos=index(line,".");
    rtpos=index(line,"real time");
    timedata=left(substr(line,rtpos+9,dotpos-(rtpos+9)));
    substr(template,8-length(timedata)+1,length(timedata))=timedata;
    time=input(template,hhmmss8.);
    output; stop;
end;
format time time8.;
```

Metadata Macros

Metadata macros:

```
%macro varexist(dataset,variable);  
  %let dsid=%sysfunc(open(&dataset)) ;  
  %sysfunc(varnum(&dsid,&variable))  
  %let dsid=%sysfunc(close(&dsid)) ;  
%mend varexist;  
  
%if %varexist(&dset,&CHAR1)>0 %then %do;  
  
%macro varlabel(dataset,variable) ;  
  %let dsid=%sysfunc(open(&dataset)) ;  
  %let varnum=%sysfunc(varnum(&dsid,&variable)) ;  
  %sysfunc(varlabel(&dsid,&varnum))  
  %let dsid=%sysfunc(close(&dsid)) ;  
%mend varlabel ;  
  
retain col1 "%varlabel(&srcdata,&linevar)";
```

↳

```
%macro vartype(dataset,variable) ;  
  %let dsid=%sysfunc(open(&dataset)) ;  
  %let varnum=%sysfunc(varnum(&dsid,&variable)) ;  
  %sysfunc(vartype(&dsid,&varnum))  
  %let dsid=%sysfunc(close(&dsid)) ;  
%mend vartype ;  
  
%if %vartype(&out,&mainvar)=N %then %do;  
  
%macro varfmtname(dataset,variable) ;  
  %let dsid=%sysfunc(open(&dataset)) ;  
  %let varnum=%sysfunc(varnum(&dsid,&variable)) ;  
  %sysfunc(varfmt(&dsid,&varnum))  
  %let dsid=%sysfunc(close(&dsid)) ;  
%mend varfmtname ;  
  
char=put(&linevar,%varfmtname(&grpdata,&linevar));
```

Conclusion

- Challenge of an epidemiology study
- Diverse elements of data
- Diverse skills
- Potential of an epidemiology study

Any Questions?

Thank You.

**Paul Murray
FTP Software Consultants Ltd.
UK**