

I See de Codes: Using SAS® to Process and Analyze ICD-9 and ICD-10 Diagnosis Codes Found in Administrative Healthcare Data

Kathy Hardis Fraeman, Evidera, Bethesda, MD

ABSTRACT

Administrative healthcare data – including insurance claims data, electronic medical records (EMR) data, and hospitalization data – contain standardized diagnosis codes used to identify diseases and other medical conditions. These codes go by their short-form name of “International Classification of Diseases,” also known as ICD. Much of the currently available healthcare data contain the 9th version of these codes, referred to as ICD-9, while the more recent 10th version ICD-10 are becoming more common in healthcare data. These diagnosis codes are typically saved as character variables, often stored in arrays of multiple codes representing primary and secondary diagnoses, and can be associated with either outpatient medical visits or inpatient hospitalizations. SAS text processing functions, array processing, and the SAS colon modifier can be used to analyze the text of these codes and identify similar codes, or ranges of ICD codes. In epidemiologic analyses, groups of multiple ICD diagnosis codes are typically used to define more general comorbidities or medical outcomes. These disease definitions based on multiple ICD diagnosis codes, also known as “coding algorithms,” can either be “hard-coded” within a SAS program, or defined externally from the programming. When coding algorithm definitions based on ICD codes are stored externally, the definitions can be read into SAS, transformed to SAS format, and dynamically converted into SAS programming statements required to identify patients with the comorbidities and outcomes of interest.

INTRODUCTION

The International Classification of Diseases (ICD) is maintained by the World Health Organization (WHO) and is described by the WHO as the international “...standard diagnostic tool for epidemiology, health management and clinical purposes.” ICD diagnosis codes provide a standardized system for classifying diseases, and also for classifying a variety of signs, symptoms, and external causes of injury or disease. These diagnosis codes are used administratively for medical reimbursement but are also used for health-related research. Researchers use ICD diagnosis codes to derive morbidity and mortality statistics, including the incidence and prevalence of disease. Because one of the main uses of these standardized diagnosis codes is for medical reimbursement, the sources of ICD diagnosis codes for research purposes are frequently found in administrative health care data – including insurance claims data, electronic medical records (EMR) data, and detailed hospitalization billing records.

WHAT DO ICD DIAGNOSIS CODES LOOK LIKE?

Much of the currently available healthcare data contain the 9th version of these codes, referred to as ICD-9, while the more recent 10th version ICD-10 released in October, 2015 are becoming more common in healthcare data. While the ICD-9 diagnosis code set defined approximately 13,000 different diagnosis codes, ICD-10 has defined approximately 68,000 different codes with the flexibility to add more new diagnosis codes as need.

Both ICD-9 and ICD-10 diagnosis codes need to be stored as SAS character variables. The character components of ICD-9 and ICD-10 diagnosis codes are different, and the resulting SAS programming required to process and analyze the different codes will reflect these differences. The general differences between the two diagnosis code sets are summarized in Table 1.

Table 1. Characteristics of ICD-9 and ICD-10 Diagnosis Codes

SAS Variable Characteristic	ICD-9 Diagnosis Codes	ICD-10 Diagnosis Codes
Information-based length	3 to 5 characters (plus additional optional decimal point between the 3 rd and 4 th characters)	3 to 7 characters (plus additional optional decimal point)
Maximum SAS character length	6 (5 information characters, plus optional decimal point)	8 (7 information characters, plus optional decimal point)
Format of first character	Numeric or alpha (only “E” or “V”)	Alpha only
Format of remaining characters	All numeric	Character 2 through 3 -- Numeric Characters 4 through 7 -- Numeric or alpha
Diagnosis code structure	Characters 1 through 3 – Disease category Optional decimal point “.” after character 3 Character 4 – Etiology, anatomic site, or disease manifestation Character 5 – Additional clinical detail	Characters 1 through 3 – Disease category Optional decimal point “.” after character 3 Characters 4 through 6 – Etiology, anatomic site, severity, or other clinical detail Character 7 – Extension, primarily used to document episodes of care for injuries
Diagnosis code examples, including optional decimal points	Sample ICD-9 diagnosis codes for pancreatic cancer 157 – Malignant neoplasm of pancreas 157.0 – Head of pancreas 157.1 – Body of pancreas 157.2 – Tail of pancreas 157.8 – Other specified sites of pancreas 157.9 – Pancreas, part unspecified	Sample ICD-10 diagnosis codes for forearm fracture S52 – Fracture of forearm S52.5 – Fracture of lower end of radius S52.52 – Torus fracture of lower end of radius S52.521 – Torus fracture of lower end of right radius S52.521A – Torus fracture of lower end or right radius, initial encounter

Because the first three characters of both ICD-9 and ICD-10 diagnosis codes represent the general disease category, and because these first three codes can overlap for codes beginning with “E” or “V”, ICD-9 and ICD-10 diagnosis codes need to either be stored in different variables or have an associated variable for diagnosis code type to differentiate between the two types of diagnosis codes..

HOW ARE ICD CODES STORED IN ADMINISTRATIVE HEALTH DATA?

In a single observation in a SAS data set, diagnosis codes can be stored in either a single variable or in multiple diagnosis code variables that can be turned into a SAS array. If an observation has multiple diagnosis code variables, the first variable corresponds to what is known as the “primary” diagnosis, and subsequent codes correspond to “secondary” diagnoses. An “admitting” diagnosis is the diagnosis used for hospital admission, and a “discharge” diagnosis is the primary diagnosis determined upon hospitalization discharge. Table 2 summarizes how diagnosis code variables are typically stored in different types of administrative healthcare data.

Table 2. ICD Diagnosis Codes Found in Administrative Health Data

Type of Healthcare Data	Type of Data Set	Typical Diagnosis Code Variables
Insurance Claims Data	Outpatient Insurance Claims	-Single outpatient diagnosis code , or -Multiple outpatient diagnosis codes , where the first code represents the primary diagnosis associated with the claim
	Inpatient Insurance Claims, Detailed (one insurance claim per inpatient billing charge)	-Single inpatient diagnosis code , or -Multiple inpatient diagnosis codes , where the first code represents the primary diagnosis associated with the claim
	Inpatient Insurance Claims, Summary (one claim per entire inpatient hospitalization)	-Multiple inpatient diagnosis codes , where the first code represents the primary diagnosis associated with the hospitalization -Admitting diagnosis -Discharge diagnosis
Electronic Medical Records (EMR) Data	Medical History	Typically single diagnosis codes per observation,
	Current Diagnoses	-Single outpatient diagnosis code , or -Multiple outpatient diagnosis codes , where the first code represents the primary diagnosis
Hospitalization Billing Data	Diagnosis Code Data	-Multiple inpatient diagnosis codes , where the first code represents the primary diagnosis -Admitting diagnosis -Discharge diagnosis

USING SAS TO ANALYZE ICD CODES

SAS text processing functions, the SAS colon modifier, and SAS array processing can be used to analyze ICD diagnosis code variables.

SAS TEXT PROCESSING FUNCTIONS

Because ICD diagnosis codes must be stored as text data, a variety of SAS text processing functions can be used to process diagnosis code variables.

The optional decimal points found in diagnosis code variables can be removed using a variety of SAS text functions, including COMPRESS and TRANWRD. Because the use of these embedded decimal points is not always consistent, it is recommended to remove them when they exist. The following is an example using COMPRESS to remove the embedded decimal in a diagnosis code variable:

```
/*-----*/
/* Remove embedded decimal point from ICD-9 code variable
/*-----*/
length diag $ 6;
diag = compress(diag, ".");
```

Sometimes components of a diagnosis code need to be processed to identify diagnosis codes of interest for research purposes. If the component of the diagnosis code does not occur at the beginning of the code, then the SUBSTR function can be used. The example below shows how the fifth character in an ICD-9 diagnosis code variable is used to identify ischemic stroke.

```
/*-----*/
/* Identify ICD-9 diagnosis codes for ischemic stroke
/*-----*/
label isch_flag = "Diagnosis code for ischemic stroke";
isch_flag = 0;
if diag in: ("433", "434") and substr(diag, 5,1) = "1" then isch_flag = 1;
```

THE SAS COLON MODIFIER

The SAS colon modifier, used as either “=” or “in:”, is a very handy tool for analyzing diagnosis codes, especially because researchers are usually concerned with the first 3 or 4 characters of a diagnosis code variable. When used after “=” or “in:”, the colon modifier looks at only the beginning values of a character variable and eliminates the need to use the SUBSTR function.

The colon modifier is used in the example above.

```
if diag in: ("433", "434") and substr(diag,5,1) = "1" then isch_flag = 1;
```

Without the colon modifier, the example above would need to have been written as:

```
if substr(diag,1,3) in ("433", "434") and substr(diag,5,1) = "1" then isch_flag = 1;
```

The use of the colon modifier becomes especially useful when the first characters of interest of a diagnosis code can be either 3 or 4 characters long.

```
/*-----*/
/* Identify ICD-9 diagnosis codes for cerebrovascular event (CVE)
/*-----*/
label = cve_flag = "Diagnosis code for cerebrovascular event";
cve_flag = 0;
if diag in: ("3466", "3623", "431", "432", "433", "434", "436") then cve_flag = 1;
```

Without using the colon modifier, the above code would need to have been written with more code using SUBSTR as:

```
if substr(diag,1,4) in ("3466", "3623") or
   substr(diag,1,3) in ("431", "432", "433", "434", "436") then cve_flag = 1;
```

The SAS colon modifier is also useful when looking for ranges of diagnosis code values. For example, ICD-9 diagnosis codes representing malignancies include all codes beginning with “140” through “208”, and “2091” through “2093” with any characters following the beginning codes. The colon

```
/*-----*/
/* Identify ICD-9 diagnosis codes for malignant neoplasms
/*-----*/
label = neo_flag = "Diagnosis code for malignant neoplasm";
neo_flag = 0;

if ("140" <=: diag <=: "208") or ("2090" <=: diag <=: "2093")
   then neo_flag = 1;
```

Similar code for ranges of ICD-10 codes would look like this:

```
/*-----*/
/* Identify ICD-10 diagnosis codes for malignant neoplasms
/*-----*/
if "C00" <=: diag <=: "C96" then neo_flag = 1;
```

SAS ARRAY PROCESSING

When multiple diagnosis codes variables are included in a SAS data set, they can be processed in arrays. The following programming examples show how SAS array processing can be used to analyze multiple diagnosis codes in one observation:

```
/*-----*/
/* Identify ICD diagnosis codes for acute myocardial infarction (MI)
/*-----*/
array diagvar $ dx1-dx4;
label mi_flag = "Diagnosis code for acute myocardial infarction";
mi_flag = 0;

do over diagvar;

    /*-----*/
    /* Acute Myocardial Infarction -- ICD-9
    /*-----*/
    if diagvar =: "410" then mi_flag = 1;

end;
```

For ICD-10 diagnosis codes, the array processing would look like this:

```
do over diagvar;

    /*-----*/
    /* Acute Myocardial Infarction -- ICD-10
    /*-----*/
    if diagvar in: ("I21", "I22") then mi_flag = 1;

end;
```

If multiple diagnosis codes are stored on the same SAS observation, but the programming rules require that only the “primary” diagnosis code be analyzed, then the SAS program would simply look at the first diagnosis code variable, in the above example the first diagnosis code variable is named DX1.

ICD DIAGNOSIS CODES IN ‘CODING ALGORITHMS’

In epidemiologic analyses, groups of multiple ICD diagnosis codes are typically used to define more general comorbidities or medical outcomes. These disease definitions based on multiple ICD diagnosis codes are known as “coding algorithms.” Some typical coding algorithms using ICD-9 and ICD-10 diagnosis codes are given in Table 3.

Table 3. Sample Diagnosis Code ‘Coding ‘Algorithms’

Medical Condition	ICD-9 Diagnosis Codes	ICD-9 Description	ICD-10 Diagnosis Codes	ICD-10 Description
Atrial Fibrillation (AF)	427.31	Atrial fibrillation	I48.x	Atrial fibrillation and flutter
Transient Ischemic Attack (TIA)	435.x	Transient cerebral ischemia	G45.8x	Other transient cerebral ischemic attacks and related syndromes
			G45.9x	Transient cerebral ischemic attack, unspecified
Pulmonary Embolism (PE)	415.11	Pulmonary embolism and infarction, iatrogenic	I26.92x	Saddle embolus of pulmonary artery without acute cor pulmonale
	415.13	Saddle embolus of pulmonary artery	I26.99x	Other pulmonary embolism without acute cor pulmonale
	415.19	Pulmonary embolism and infarction, other		

USING SAS TO IMPLEMENT EXTERNALLY DEFINED CODING ALGORITHMS

Coding algorithms using combinations of ICD diagnosis codes, similar to the examples shown in Table 3, are frequently used in health research studies. These algorithms are often unique to each study and frequently change during study development as new diagnosis codes are added, or existing diagnosis codes are modified or deleted. Although the SAS programmer is responsible for getting these coding algorithms in to the SAS programs and for ensuring the coding algorithm definitions remain current and up-to-date, the non-programming study scientists and researchers are usually the ones who define these coding algorithms and who make any subsequent modifications.

To aid in this effort, a study’s code definitions can be stored externally, typically in an Excel spreadsheet. Storing the coding algorithm definitions in an external data source have many advantages. Non-programmers can clearly understand the coding algorithm definitions, and the external data can be converted into SAS data and then dynamically converted into SAS code each time the definitions are changed. Through Excel-to-SAS conversion and data manipulation programming, the information stored in Table 4 could be converted into the SAS data set shown below.

Obs	condition	abrv	code	codeset
1	Atrial Fibrillation	AF	427.31	9
2	Atrial Fibrillation	AF	I48.x	10
3	Transient Ischemic Attack	TIA	435.x	9
4	Transient Ischemic Attack	TIA	G45.8x	10
5	Transient Ischemic Attack	TIA	G45.9x	10
6	Pulmonary Embolism	PE	415.11	9
7	Pulmonary Embolism	PE	415.13	9
8	Pulmonary Embolism	PE	415.19	9
9	Pulmonary Embolism	PE	I26.92x	10
10	Pulmonary Embolism	PE	I26.99x	10

The SAS data set containing the ICD diagnosis codes in a coding algorithm can be converted into SAS code in many ways. The example below shows how to put the different groups of ICD diagnosis codes into global macro variables using PROC SQL.

```

/*****
/* Take out the decimal point and wild card "x"
*****/
data CODES1;
    set CODES;
    code = compress(code, ".x");
run;

/*****
/* Assign diagnosis codes to global macro variables
*****/
%global af9 af10 tia9 tia10 pe9 pe10;

proc sql noprint;
    /*-----*/
    /* Atrial Fibrillation
    /*-----*/
    select quote(trim(code))
    into :af9 separated by ', '
    from codes1
    where codeset = 9 and abrv = "AF";
    select quote(trim(code))
    into :af10 separated by ', '
    from codes1
    where codeset = 10 and abrv = "AF";

    /*-----*/
    /* Transient Ischemic Attack
    /*-----*/
    select quote(trim(code))
    into :tia9 separated by ', '
    from codes1
    where codeset = 9 and abrv = "TIA";
    select quote(trim(code))
    into :tia10 separated by ', '
    from codes1
    where codeset = 10 and abrv = "TIA";

    /*-----*/
    /* Pulmonary Embolism
    /*-----*/
    select quote(trim(code))
    into :pe9 separated by ', '
    from codes1
    where codeset = 9 and abrv = "PE";
    select quote(trim(code))
    into :pe10 separated by ', '
    from codes1
    where codeset = 10 and abrv = "PE";

quit;

%put &af9;
%put &af10;
%put &tia9;
%put &tia10;
%put &pe9;
%put &pe10;

```

The macro variables containing the list of ICD diagnosis codes are shown below.

```
201 %put &af9;
"42731"
202 %put &af10;
"I48"
203 %put &tia9;
"435"
204 %put &tia10;
"G458", "G459"
205 %put &pe9;
"41511", "41513", "41519"
206 %put &pe10;
"I2692", "I2699"
```

These disease definitions stored in macro variables can then be used in programming that identifies diagnosis codes in insurance claims data. If the disease definitions change in the external data file, such as an Excel file, then the programming will be automatically updated to the most recent definitions.

```
data claims_flag;
    set claims_orig;

    label af_flag = "Atrial fibrillation diagnosis code"
          tia_flag = "Transient ischemic attack diagnosis code"
          pe_flag = "Pulmonary Embolism"
          ;

    af_flag = 0;
    tia_flag = 0;
    pe_flag = 0;

    array diagvar $ dx1-dx4;

    do over diagvar;
        if dx_version = 9 then do;
            if diagvar in: (&af9) then af_flag = 1;
            if diagvar in: (&tia9) then tia_flag = 1;
            if diagvar in: (&pe9) then pe_flag = 1;
        end;
        else if dx_version = 10 then do;
            if diagvar in: (&af10) then af_flag = 1;
            if diagvar in: (&tia10) then tia_flag = 1;
            if diagvar in: (&pe10) then pe_flag = 1;
        end;
    end;

run;
```

CONCLUSION

ICD diagnosis codes are available in many types of administrative healthcare data, and are a valuable tool for health-related research. SAS provides many programming tools for the processing and analysis of these codes. Disease definitions based on multiple ICD diagnosis codes, can either be hard-coded within a SAS program, or defined externally outside the programming. When coding algorithm definitions based on ICD codes are stored externally, the definitions can be read into SAS, transformed to SAS format, and dynamically converted into SAS programming statements required to identify patients with the comorbidities and outcomes of interest.

REFERENCES

World Health Organization, "International Classification of Diseases (ICD) Information Sheet", found at
<http://www.who.int/classifications/icd/factsheet/en/>

Kathy Hardis Fraeman, "Let SAS® Write and Execute Your Data-Driven SAS Code", NESUG 2010, found at
<http://www.lexjansen.com/nesug/nesug10/cc/cc12.pdf>

Jason Ford, "Introduction to Character String Functions", NESUG 2010, found at
<http://www.lexjansen.com/nesug/nesug10/ff/ff06.pdf>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Kathy H. Fraeman
Director Data Analytics
Real World Evidence
Evidera
7101 Wisconsin Ave #1400
Bethesda, MD 20814
Work Phone: +1 240 235 2525
Fax: +1 301 654 9864
E-mail: kathy.fraeman@evidera.com
Web: www.evidera.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.