

SAS[®] Programming

with Medicare Administrative Data



Matthew Gillingham



From *SAS® Programming with Medicare Administrative Data*.
Full book available for purchase [here](#).

Contents

About This Book	ix
About The Author	xiii
Acknowledgments	xv
Chapter 1: Introduction	1
Introduction and Purpose of This Book	1
Framework of This Book	2
Our Programming Project	3
Chapter Outline	5
How to Use This Book	6
Disclaimer	7
Chapter Summary	8
Chapter 2: An Introduction to the Medicare Program	9
Introduction and Goals of the Chapter	9
An Introduction to the Medicare Program	10
What is Medicare?	10
Medicare Enrollment and Eligibility	11
What Is Covered by Medicare?	11
What Is Not Covered by Medicare?	12
The Mechanics of Medicare	13
Chapter Summary	15
Chapter 3: An Introduction to Medicare Data	17
Introduction and Goals of This Chapter	17
A Note on the Source and Structure of Our Claims Data	18
Part B Carrier Claims Data	19
Durable Medical Equipment (DME) Data	19

Outpatient Claims Data	20
Inpatient Claims Data	20
Skilled Nursing Facility (SNF) Claims Data.....	21
Home Health Claims Data.....	21
Hospice Claims Data	22
Commonly Retained Elements in Administrative Claims Data	22
Master Beneficiary Summary File	24
Provider Data.....	24
Example: Identification of Emergency Department (ED) Utilization.....	25
Example: Identification of Surgical Services	26
Chapter Summary.....	26
Chapter 4: Planning the Research Programming Project.....	29
Introduction and Goals of This Chapter	29
Our Project's Plan: The SDLC	30
Our Project's High-Level Requirements.....	32
Our Project's Quality Assurance Plan	33
Our Project's Flowchart	33
Our Project's Data Storage and LIBNAMES	35
Our Research Programming Technical Specifications.....	36
Chapter Summary.....	37
Chapter 5: Request, Receive, Load, and Transform Data	39
Introduction	39
Requesting Medicare Administrative Claims and Enrollment Data	40
Sources of Medicare Claims and Enrollment Data	40
Data Available from CMS	41
Our Project's Data Requirements and Request Specifications.....	42
Contacting ResDAC and Completing Paperwork	43
Receiving, Decrypting, and Loading Medicare Administrative Claims and Enrollment Data	45
Receiving Our Medicare Administrative Data	45
Decrypting Our Medicare Administrative Data	46
Loading Our Medicare Administrative Data into SAS Data Sets.....	46
Algorithms: Transforming Base Claim and Line Level Data Sets into Single Claim-Level	

Files.....	47
Transforming Base Claim and Line-Level Carrier Data into a Claim-Level File	48
Chapter Summary	51
Chapter 6: Working with Enrollment Data	53
Introduction and Goals	53
Review and Approach.....	54
Basics of Medicare Enrollment Data	54
Our Programming Plan.....	55
Algorithms: Identifying Continuously Enrolled FFS Beneficiaries.....	55
Why Define Continuously Enrolled FFS Beneficiaries?	55
How to Specify the Programming for Continuous Enrollment in Medicare FFS	56
Medicare Part A and Part B Enrollment Variables	56
HMO Coverage Variable.....	56
Date of Death Variable	57
How to Program in SAS to Define Continuous Enrollment in Medicare FFS	57
Algorithms: Create or Retain Data Elements for Displaying Results by Certain Characteristics	59
Coverage Characteristics, Month of Death, Sex, and Race.....	59
Age Groups.....	62
Geographic Characteristics.....	63
Algorithms: Create Final Enrollment Data for Remainder of Programming	66
Chapter Summary	67
Exercises	67
Chapter 7: Measuring Utilization of Services	69
Introduction and Goals	69
Review and Approach.....	70
Review: Basics of Medicare Claims Data	70
Our Programming Plan.....	72
Algorithms: Delimit Claims by Continuously Enrolled Beneficiary Population.....	72
Algorithms: Measuring Evaluation and Management (E&M) Utilization.....	73
Algorithms: Measuring Inpatient Hospital Utilization.....	75
Algorithms: Measuring Emergency Department Utilization	80
Algorithms: Measuring Utilization of Ambulance Services.....	81
Algorithms: Measuring Outpatient Visit Information	84
Algorithms: Measuring Utilization of SNF, Home Health Agency, and Hospice Care	85

Chapter Summary	88
Exercises	88
Chapter 8: Measuring Costs to Medicare	91
Introduction and Goals.....	91
Review and Approach	93
Review: Basics of Medicare Cost.....	93
Our Programming Plan	93
A Note on Payment Standardization and Risk Adjustment	94
Algorithms: Measuring Evaluation and Management Payments.....	95
Algorithms: Measuring Inpatient Hospital Payments	98
Algorithms: Measuring Total Part A Payments.....	100
Chapter Summary	103
Exercises	103
Chapter 9: Programming to Identify Chronic Conditions	105
Introduction and Goals.....	105
Review and Approach	107
Review: Peculiarities of Medicare Data	107
Our Programming Plan	107
A Note on the Chronic Conditions Data Warehouse (CCW)	108
Algorithms: Identifying Beneficiaries with Diabetes or COPD	108
Algorithms: Evaluation and Management Visits for Beneficiaries with Diabetes or COPD.....	114
Algorithms: Diabetic Eye Exams for Beneficiaries with Diabetes	116
Algorithms: Hospital Readmissions for Beneficiaries with COPD	118
Chapter Summary	120
Exercises	121
Chapter 10: Presenting Output and Project Disposition	123
Introduction	123
Synthesis – Review and Approach.....	124
Review: Our Accomplishments.....	124
Our Programming Plan	124
Algorithms: Presenting Selected Measurements of Utilization, Cost to Medicare, and	

Quality Outcomes	125
Presenting Ambulance Utilization	125
Presenting Medicare Payments for Evaluation and Management Services by Provider	127
Presenting Rates of Diabetes and COPD	129
Algorithms: Presenting Inpatient Length of Stay Information by State	131
Algorithms: Presenting Mean Medicare Part A Payments per Beneficiary by State and County	133
Algorithms: Presenting Rates of Diabetic Eye Exams	135
Production Execution in Batch	138
Project Disposition	138
Archiving Materials for Future Use	138
Data Destruction	139
Chapter Summary	139
Exercises	140

From *SAS® Programming with Medicare Administrative Data* by Matthew Gillingham. Copyright © 2014, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



From *SAS® Programming with Medicare Administrative Data*.
Full book available for purchase [here](#).

Chapter 9: Programming to Identify Chronic Conditions

Introduction and Goals	105
Review and Approach	107
Review: Peculiarities of Medicare Data.....	107
Our Programming Plan.....	107
A Note on the Chronic Conditions Data Warehouse (CCW)	108
Algorithms: Identifying Beneficiaries with Diabetes or COPD	108
Algorithms: Evaluation and Management Visits for Beneficiaries with Diabetes or COPD	114
Algorithms: Diabetic Eye Exams for Beneficiaries with Diabetes	116
Algorithms: Hospital Readmissions for Beneficiaries with COPD	118
Chapter Summary	120
Exercises	121

Introduction and Goals

In simple terms, chronic conditions are medical conditions which have some permanence and can progressively worsen over time, thus requiring long-term management to monitor or treat (e.g., control symptoms or otherwise influence the course of the condition). Chronic conditions are not curable by any short-term treatment, and can exhibit remissions and exacerbations.¹ Commonly identified chronic conditions in the Medicare beneficiary population include Alzheimer’s disease, asthma, chronic kidney disease, chronic obstructive pulmonary disease (COPD), depression, diabetes, glaucoma, and certain types of cancer, like breast cancer or colorectal cancer.²

Chronic conditions are important to study for several reasons. First, many Medicare beneficiaries are identified as having at least one chronic condition. For example, roughly 4 million Medicare beneficiaries are identified as having COPD and about 10 million beneficiaries are identified as diabetic.³ Second, beneficiaries with chronic conditions require a high level of specialized and concerted medical care, often resulting in the generation of many claims. In other words, utilization of services for beneficiaries with chronic conditions will be relatively higher than average, and cost may follow suit. Finally, beneficiaries identified as having chronic conditions are commonly used in the measurement of

quality outcomes because they benefit from certain standard medical practices. For example, diabetics benefit from annual dilated retinal eye exams to monitor changes in the retina (called retinopathy) that can seriously harm one's vision. Studying whether a provider performs this test on diabetic patients is a good way of measuring the quality of care provided by the doctor.

In this chapter, we will advance our discussions from Chapter 7 and Chapter 8 by providing more detail on how to measure quality outcomes. Specifically, we will combine the study of two common chronic conditions, COPD and diabetes, with the concept of measuring patient outcomes. Admittedly, quality of care and value measurement are controversial topics, especially if the measurements play a role in the assessment of provider performance or the determination of payment to providers. Indeed, patient outcomes (like hospital readmissions) often depend on things outside of providers' control, like the behavior of the patient (e.g., does the patient exercise regularly, smoke, or reliably take prescribed medications?) and the patient's general level of health when he or she began treatment. What's more, some of these factors, such as whether or not a patient exercises regularly, cannot be measured with Medicare administrative claims data.

Given the above, our goal in this chapter is not to make any determinations about quality of care; after all, we are using fake data purely for simulation purposes! Rather, our goal is to use our example research programming project to teach the reader the mechanics of how to use Medicare data to identify chronic conditions and commonly used indicators of quality outcomes. By the end of this chapter, the reader will understand important foundational concepts to use Medicare administrative claims and enrollment data to identify most any chronic condition or compute most any quality outcome metric. For example, we will discuss calculating hospital readmissions and using diagnosis and procedure codes to identify a condition or an event related to quality outcome measurement (such as diabetic eye exams). The reader could adapt the measurement of eye exams for diabetics to examine a different procedure (say, an immunization for influenza) for beneficiaries with a different chronic condition (such as heart failure). To these ends, in this chapter, we will code algorithms to do the following:

- Use our Medicare administrative claims data to identify beneficiaries with two chronic conditions, diabetes and COPD.
- Measure evaluation and management utilization for beneficiaries with diabetes or COPD.
- Measure utilization of diabetic eye exams for beneficiaries with diabetes.
- Measure hospital readmissions for beneficiaries with COPD.

As discussed in earlier chapters, the online companion to this book is at <http://support.sas.com/publishing/authors/gillingham.html>. Here, you will find information on creating dummy source data, the code in this and subsequent chapters, as well as answers to the exercises in this book. I expect you to visit the book's website, create your own dummy source data, and run the code yourself.

Review and Approach

Review: Peculiarities of Medicare Data

Refer to the reviews in Chapters 6, 7, and 8 to review the material relevant to this chapter's study of chronic conditions and quality. As mentioned above, quality measurement can be controversial because it sometimes involves rating provider performance. With this in mind, let's review some peculiarities and possible limitations of Medicare administrative data so we can be better prepared to interpret our output.

- Medicare coverage for the elderly is provided regardless of medical history. For example, a beneficiary diagnosed with COPD who becomes eligible for Medicare is granted coverage regardless of her COPD diagnosis. If you are used to working with commercial healthcare claims data, you will likely notice some unique characteristics of the Medicare population, like more beneficiaries identified as having at least one chronic condition (though this may change with the implementation of the Affordable Care Act).
- Some services may not appear in Medicare administrative data. For example, prescription drugs administered during a hospital stay may not appear in the claims data at all. Additionally, services paid for by Medicare Part C may not appear in the administrative claims files because managed care providers pay them.
- The administrative data we use for research purposes is updated on a regular basis, but only with claims that have been received, adjudicated, and deemed final action. As such, the files we use at any given time do not contain all final action claims submitted and paid up to the date of extraction of the data. In some rare cases, when not impractical, medical record reviews can fill in any gaps in information. What's more, we will see changes to the information available for quality measurement as electronic medical records become more prevalent.
- The correct composition of any study population must be determined carefully. In Chapter 6, we determined that our example research project will examine only those beneficiaries continuously enrolled in fee-for-service Medicare during a defined timeframe, a decision that has implications for the output of our example research programming project. More specifically, this step resulted in eliminating beneficiaries for whom we may not have a full set of claims, giving us more confidence that we are not missing claims for services like retinal eye exams or hospital readmissions for our study population.

Our Programming Plan

In Chapter 5, we submitted a data request to extract all claims and enrollment information for beneficiaries who were treated by the providers in our study population. We received and loaded our data, keeping only those variables we wished to use in our study. In Chapter 6, we used enrollment data to delimit our population to full-year FFS beneficiaries. In Chapter 7, we brought our claims data to bear on the calculation of utilization measurements. In Chapter 8, we used the Medicare amount paid to measure payments of specific services. In this chapter, we will draw on the study of utilization and payment by looking at two chronic conditions, diabetes and COPD. First, we will use diagnosis codes in specific settings of care to identify beneficiaries with diabetes or COPD (or both). Next, we will look at

quality of care for these beneficiaries, giving us an excellent chance to apply the programming methods we learned in Chapters 7 and 8.

The output created in this chapter will be analytic files at the beneficiary level, which we will use for the remainder of our programming work. Looking forward, in Chapter 10, we will end our example project by summarizing all of our output.

A Note on the Chronic Conditions Data Warehouse (CCW)

Readers familiar with Medicare administrative data may be trying to reconcile two facts:

- We are coding algorithms to identify beneficiaries with two chronic conditions, diabetes and COPD, using claims data.
- CMS provides files containing this information to the research community through the Chronic Condition Data Warehouse (CCW). The CCW “is a research database designed to make Medicare, Medicaid, Assessments, and Part D Prescription Drug Event data more readily available to support research designed to improve the quality of care and reduce costs and utilization.”⁴ These files are available through ResDAC and CMS’s data distribution contractor.

With some simplifications, we will draw on the CCW’s definitions of diabetes and COPD for coding the identification of diabetes and COPD in our beneficiary population. But why do we code these algorithms ourselves when flags identifying beneficiaries with certain chronic conditions are provided by CMS? Coding chronic conditions is useful for instructional purposes, providing a context for understanding and modifying definitions of chronic conditions not available in the CCW. In addition, some investigators choose to define chronic conditions because they wish to modify the definition used in the CCW. For example, the CCW definition of diabetes includes secondary diabetes, which may not be the right choice for your particular study. It should go without saying that readers are encouraged to take full advantage of the data offered in the CCW.

Algorithms: Identifying Beneficiaries with Diabetes or COPD

Our first task is to identify beneficiaries in our population with diabetes or COPD. At the risk of being repetitive, before we begin, it is very important to note that the diagnosis codes we will use can change from year to year. The same goes for any of the codes we use in this book. For example, procedure and diagnosis codes are updated annually (new codes are created and some codes are retired). Therefore, it is good procedure to regularly check the codes you use to identify diagnoses or services (or anything, for that matter) to ensure they are current. This will be especially true after October 2014, as all codes will change when a new diagnosis coding system, ICD-10-CM, is implemented.

The algorithm to identify beneficiaries with diabetes is adapted from instructions available from the CCW, with some simplifications (e.g., for the identification of diabetes we do not require looking across two years of data).⁵ Our SAS code involves searching for specific diagnosis codes (see the below algorithms for the lists of specific diagnosis codes). These codes can occur at any position on the claim (i.e., they do not have to be primary diagnoses). The setting in which these diagnosis codes occurs is

important. Specifically, if we find evidence that a beneficiary was diagnosed with diabetes on only one occasion in an inpatient or skilled nursing facility setting, we will count that beneficiary as having diabetes. In these settings, trained coding experts review medical records, consult with physicians, and only record a diagnosis if the patient actually has the condition. However, we need two separate events of a diagnosis of diabetes when searching in carrier and hospital outpatient claims.⁶ Why? Because the occurrence of a single diagnosis of diabetes in these settings may be what is called a “rule out” diagnosis, where the physician is ordering a lab test to check for diabetes. The second diagnosis confirms the beneficiary does have diabetes. In all cases, the diagnoses can occur any time during a two-year period (but we will search just one year of data in our example). Finally, the reader will note that we use home health claims data, but not hospice or DME data, in our identification of beneficiaries with diabetes or COPD. We use home health data even though investigators view diagnosis codes on home health claims as somewhat unreliable. We do not use hospice claims due to the nature of care provided in a hospice setting. Lastly, for obvious reasons, the DME file cannot be used to reliably identify beneficiaries with diabetes or COPD.

The algorithm to identify beneficiaries with COPD is constructed in a similar manner (and also adapted from information provided by the CCW) to the algorithm to identify beneficiaries with diabetes, but using different diagnosis codes (and the search typically occurs over a one-year period of claims data). Therefore, we will code a single algorithm to search inpatient, SNF, outpatient, and Part B carrier claims for evidence of these diagnosis codes, setting flags where appropriate. Then, we will count the flags we created, and use these counts to identify beneficiaries with diabetes or COPD.

In Step 9.1, we present an algorithm for identifying the presence of diagnosis codes used to identify diabetes or COPD in our inpatient, SNF, home health, and outpatient claims data sets. This algorithm is similar to code used in previous chapters to identify diagnosis codes in claims data. We use a macro called `CLM_LOOP` and array logic to create separate flags set equal to 1 in the presence of diabetes (`DIA_IP`, `DIA_SN`, `DIA_OP`, and `DIA_HH`) and COPD (`COPD_IP`, `COPD_SN`, `COPD_OP`, and `COPD_HH`) on a particular claim. Each of these claims data sets contain 25 diagnosis code variables (`ICD_DGNS_CD1-ICD_DGNS_CD25`). In addition, we must use conditional processing for our inpatient claims data set because, again, we are only searching claims for inpatient short stay and critical access hospitals with non-negative values of `PMT_AMT`. Therefore, we use a `%if` clause to read the inpatient claims in `UTL.IP_2010_FNL_SS`. We output only those records where the flag variables are equal to 1. In Step 9.3 and Step 9.4, we will use these records as definitive proof that the beneficiary has diabetes or COPD (with the exception of the flags derived from outpatient claims data).

```

/* STEP 9.1: SEARCH INPATIENT, SNF, HOME HEALTH, AND OUTPATIENT CLAIMS
FOR DIABETES AND COPD DIAGNOSIS CODES */
/* NOTE THAT THIS MACRO WILL ERROR OUT IF CALLED FOR A CLMTYP VALUE WE
DID NOT CREATE */
%macro clm_loop(clmtyp= );
data cc_&clmtyp._search(keep=bene_id dia_&clmtyp. copd_&clmtyp.);
  %if &clmtyp.=sn or &clmtyp.=op or &clmtyp.=hh %then %do;
    set utl.&clmtyp._2010_fnl;
  %end;
  %else %if &clmtyp.=ip %then %do;
    set utl.ip_2010_fnl_ss;
  %end;
endmacro;

```

```

    %end;
    array dgnsacd(25) icd_dgns_cd1-icd_dgns_cd25;
    do i=1 to 25;
        /* diabetes */
        if substrn(dgnsacd(i),1,4)='3572'
            or substrn(dgnsacd(i),1,5)
in('24900','24901','24910','24911','24920','24921','24930','24931',
    '24940','24941','24950','24951','24960','24961','24970','24971','
24980','24981','24990','24991',
    '25000','25001','25002','25003','25010','25011','25012','25013','
25020','25021','25022','25023',
    '25030','25031','25032','25033','25040','25041','25042','25043','
25050','25051','25052','25053',
    '25060','25061','25062','25063','25070','25071','25072','25073','
25080','25081','25082','25083',
    '25090','25091','25092','25093','36201','36202','36641')
then dia_&clmtyp.=1;
        /* copd */
        if substrn(dgnsacd(i),1,3)='496'
            or substrn(dgnsacd(i),1,4)
in('4910','4911','4918','4919','4920','4928','4940','4941')
            or substrn(dgnsacd(i),1,5) in('49120','49121','49122') then
copd_&clmtyp.=1;
        end;
        if dia_&clmtyp.=1 or copd_&clmtyp.=1;
    run;
%mend clm_loop;
%clm_loop(clmtyp=ip);
%clm_loop(clmtyp=sn);
%clm_loop(clmtyp=op);
%clm_loop(clmtyp=hh);

```

In Step 9.2, we continue the process of identifying the presence of diagnosis codes used to identify diabetes or COPD, this time using our carrier claims data set. Once again, we search each claim for diagnosis codes that indicate the presence of diabetes or COPD. When such a diagnosis code value is found in one of the carrier file's 13 line-level⁷ diagnosis codes (**LINE_ICD_DGNS_CD1-LINE_ICD_DGNS_CD13**), we set the relevant flag (**DIA_CARR**, **COPD_CARR**, or both) equal to 1. We output only those records where the flag variables are equal to 1. In Step 9.3 and Step 9.4, we will use these records to build proof that the beneficiary has diabetes or COPD.

```

/* STEP 9.2: SEARCH CARRIER CLAIMS FOR DIABETES AND COPD DIAGNOSIS CODES
*/
data cc_carr_search(keep=bene_id dia_carr copd_carr);
  set utl.carr_2010_fnl;
  array lndgns(13) line_icd_dgns_cd1-line_icd_dgns_cd13;
  do i=1 to 13;
    /* diabetes */
    if substrn(lndgns(i),1,4)='3572'
      or substrn(lndgns(i),1,5)
in('24900','24901','24910','24911','24920','24921','24930','24931',
    '24940','24941','24950','24951','24960','24961','24970','24971',
24980','24981','24990','24991',
    '25000','25001','25002','25003','25010','25011','25012','25013',
25020','25021','25022','25023',
    '25030','25031','25032','25033','25040','25041','25042','25043',
25050','25051','25052','25053',
    '25060','25061','25062','25063','25070','25071','25072','25073',
25080','25081','25082','25083',
    '25090','25091','25092','25093','36201','36202','36641')
  then dia_carr=1;
    /* copd */
    if substrn(lndgns(i),1,3)='496'
      or substrn(lndgns(i),1,4) in('4910 ','
4911','4918','4919','4920','4928','4940','4941')
      or substrn(lndgns(i),1,5) in('49120','49121','49122') then
  copd_carr=1;
    end;
    if dia_carr=1 or copd_carr=1;
  run;

```

In Step 9.3, we use a macro to loop (called **CLM_LOOP2**) through all claim types processed, summarizing the flags for each beneficiary by claim type, outputting one file for each relevant claim type (**CC_CNT_IP**, **CC_CNT_SN**, **CC_CNT_OP**, **CC_CNT_HH**, and **CC_CNT_CARR**). In so doing, we create summary variables called **DIA_IP_SUM**, **DIA_SN_SUM**, **DIA_OP_SUM**, **DIA_HH_SUM** and **DIA_CARR_SUM** equal to the total number of claims with the presence of a diagnosis code for diabetes in the relevant claims data set for each beneficiary. Similarly, we create summary variables called **COPD_IP_SUM**, **COPD_SN_SUM**, **COPD_OP_SUM**, **COPD_HH_SUM** and **COPD_CARR_SUM** equal to the total

number of claims with the presence of a diagnosis code for COPD in the relevant claims data set for each beneficiary. Lastly, we perform nodupkey sort of the outputted files, keeping only one record per beneficiary. We now turn our attention to using these summary variables to make a final determination on whether or not a particular beneficiary has diabetes or COPD.

```

/* STEP 9.3: COUNT OCCURRENCES OF FLAGS BY SETTING */
%macro clm_loop2(clmtyp2=);
proc sql;
    create table cc_cnt_&clmtyp2. as
    select *, sum(dia_&clmtyp2.) as dia_&clmtyp2._sum,
sum(copd_&clmtyp2.) as copd_&clmtyp2._sum
    from cc_&clmtyp2._search
    group by bene_id;
quit;

/**/proc sort data=cc_cnt_&clmtyp2. nodupkey; by bene_id; run;
%mend clm_loop2;
%clm_loop2(clmtyp2=ip);
%clm_loop2(clmtyp2=sn);
%clm_loop2(clmtyp2=op);
%clm_loop2(clmtyp2=hh);
%clm_loop2(clmtyp2=carr);

```

In Step 9.4, we see the above explanation of how to count diagnosis codes for a particular claim type come to fruition. Merging the output of Step 9.3 into a single data set, the code in this step uses the summarized diabetes and COPD flags created in Step 9.3 to identify beneficiaries with diabetes or COPD as follows:

- A beneficiary is identified as having diabetes in the presence of a single relevant diagnosis found in inpatient, SNF, or home health claims data. This evidence is found where **DIA_IP_SUM**, **DIA_SN_SUM**, or **DIA_HH_SUM** is greater than or equal to one.
- A beneficiary is also identified as having diabetes in the presence of two or more relevant diagnoses found on separate outpatient or carrier claims. This evidence is found where **DIA_OP_SUM** or **DIA_CARR_SUM** is greater than or equal to two.
- A beneficiary is identified as having COPD in the presence of a single relevant diagnosis found in inpatient, SNF, or home health claims data. This evidence is found where **COPD_IP_SUM**, **COPD_SN_SUM**, or **COPD_HH_SUM** is greater than or equal to one.

- A beneficiary is also identified as having COPD in the presence of two or more relevant diagnoses found on separate outpatient or carrier claims. This evidence is found where **COPD_OP_SUM** or **COPD_CARR_SUM** is greater than or equal to two.

```

/* STEP 9.4: IDENTIFY BENEFICIARIES WITH DIABETES OR COPD USING SUMMED
FLAGS */
data cnd.dia_copd_cond(keep=bene_id dia_flag copd_flag);
    merge cc_cnt_ip cc_cnt_sn cc_cnt_op cc_cnt_hh cc_cnt_carr;
    by bene_id;
    if dia_ip_sum>=1 or dia_sn_sum>=1 or dia_hh_sum>=1 or
dia_op_sum>=2 or dia_carr_sum>=2 then dia_flag=1; else dia_flag=0;
    if copd_ip_sum>=1 or copd_sn_sum>=1 or copd_hh_sum>=1 or
copd_op_sum>=2 or copd_carr_sum>=2 then copd_flag=1; else copd_flag=0;
    if dia_flag=1 or copd_flag=1;
run;

proc sort data=cnd.dia_copd_cond nodupkey;
    by bene_id;
run;

```

Beneficiaries with diabetes are assigned a value of 1 to the **DIA_FLAG** variable, and beneficiaries with COPD are assigned a value of 1 to the **COPD_FLAG** variable. We then de-duplicate our output data set by beneficiary. We have now identified our subpopulation of beneficiaries with diabetes or COPD, contained in the outputted data set called **CND.DIA_COPD_COND**. Output 9.1 displays a print of the first 10 observations in the **CND.DIA_COPD_COND** data set.⁸

Output 9.1: Print of Beneficiaries with Diabetes or COPD***PRINT OF BENEFICIARIES WITH DIABETES OR COPD***

Obs	BENE_ID	dia_flag	copd_flag
1	0004F0ABD505251D	1	0
2	00052705243EA128	1	0
3	00070B63745BE497	1	0
4	0007F12A492FD25D	1	1
5	00139C345A104F72	0	1
6	0013E139F1F37264	1	0
7	00157F1570C74E09	1	0
8	001876C5D3F3F6AA	1	0
9	0018BD6F2F493452	1	0
10	00196F0702489342	1	0

Algorithms: Evaluation and Management Visits for Beneficiaries with Diabetes or COPD

At this point in our project, we have both a list of beneficiaries in our population with diabetes or COPD, and a data set describing the E&M utilization of beneficiaries in our population. Indeed, the data set of E&M utilization called UTL.EM_UTIL, developed in Steps 7.2 through 7.4 of the code created in Chapter 7, contains a list of E&M services for the beneficiaries in our population. Our algorithm developed in Step 9.5 acknowledges that it is easier simply to merge our list of beneficiaries with COPD or diabetes with this E&M utilization data set created in Chapter 7. The output of this step is a data set

called CND.EM_COND, and contains a listing of E&M services dates for beneficiaries with diabetes or COPD. To round out our work in Step 9.5, we summarize the **EM_SVC** flag for each beneficiary using a PROC FREQ, and use SAS' Output Delivery System (ODS) to create our results.

```
/* STEP 9.5: MERGE E&M UTILIZATION DATA WITH LIST OF BENEFICIARIES WITH
DIABETES OR COPD */
data cnd.em_cond;
    merge cnd.dia_copd_cond(in=a) utl.em_util;
    by bene_id;
    if a;
run;

ods html file="C:\Users\mgillingham\Desktop\SAS
Book\FINAL_DATA\ODS_OUTPUT\Gillingham_fig9_5.html"
image_dpi=300 style=GrayscalePrinter;
ods graphics on / imagefmt=png;
title "FREQ OF EM SERVICES FOR BENEFICIARIES WITH DIABETES OR COPD";
proc freq data=cnd.em_cond;
    tables em_svc*dia_flag*copd_flag;
run;
ods html close;
```

Output 9.2 shows the frequency distribution run in Step 9.5.

Output 9.2: Frequency of EM Services

*FREQ OF EM SERVICES FOR BENEFICIARIES WITH DIABETES OR COPD**The FREQ Procedure*

Frequency Percent Row Pct Col Pct	Table 1 of dia_flag by copd_flag			
	Controlling for em_svc=1			
	dia_flag	copd_flag		
		0	1	Total
0	0	2113	2113	
	0.00	7.24	7.24	
	0.00	100.00		
	0.00	37.46		
1	23548	3528	27076	
	80.67	12.09	92.76	
	86.97	13.03		
	100.00	62.54		
Total	23548	5641	29189	
	80.67	19.33	100.00	

Algorithms: Diabetic Eye Exams for Beneficiaries with Diabetes

Now that we have a taste of using claims data to analyze services for beneficiaries with chronic conditions, let's search for evidence of eye exam services for the beneficiaries we identified as having diabetes. We will keep our example simple and use a small set of procedure codes to identify retinal eye exams (presented in the code below). However, it is possible to use additional codes, like ICD-9-CM surgical procedure codes. If we were calculating an official quality measure (e.g., an NCQA HEDIS measure), we would perform exclusions on our population of beneficiaries. However, for our purposes, a simple check of the carrier file will suffice; although it would be easy enough to apply this algorithm to other settings, like inpatient claims, using algorithms developed in this chapter.

In Step 9.6, we create a data set called CND.EYE_DIA_COND that contains the beneficiary identifier (**BENE_ID**) and a flag variable called **EYE_SVC** indicating whether we found evidence of an eye exam for this beneficiary in the carrier claims. First, we merge the data set created in Step 9.4 that identifies beneficiaries with diabetes or COPD (CND.DIA_COPD_COND) with the carrier claims data set (UTL.CARR_2010_FNL). Because we are interested in eye exams for beneficiaries identified as having diabetes, we subset the CND.DIA_COPD_COND data set with a 'where' clause specifying that we retain records where **DIA_FLAG** is equal to 1. From this point forward, it is a matter of executing code

to search the 13 line item procedure code variables for evidence of an eye exam service (this code should look very familiar!).⁹ We keep records for only those diabetic beneficiaries for whom we found evidence of an eye exam.

```

/* STEP 9.6: PULL RECORDS FROM CARRIER FILE USING CODES FOR RETINAL EYE
EXAMS */
data cnd.eye_dia_cond(keep=bene_id eye_svc);
    merge cnd.dia_copd_cond(in=a where=(dia_flag=1))
    utl.carr_2010_fnl(keep=bene_id hcpcs_cd:);
    by bene_id;
    if a;
    array hcpcscd(13) hcpcs_cd1-hcpcs_cd13;
    eye_svc=0;
    do i=1 to 13;
        if substrn(hcpcscd(i),1,5)
in('92002','92004','92012','92014','92018','92019',

        '92225','92226','92230','92235','92240','92250','92260','99203','
99204','99205','99213','99214',
        '99215','99242','99243','99244','99245') then eye_svc=1;
        leave;
    end;
    if eye_svc=1;
run;

```

In Step 9.7, we de-duplicate the output of Step 9.6, overwriting the data set called CND.EYE_DIA_COND so that it now contains just one record per beneficiary.

```

/* STEP 9.7: DEDUPLICATE THE OUTPUT OF PREVIOUS STEP TO CONTAIN ONE
RECORD PER BENEFICIARY */
proc sort data=cnd.eye_dia_cond nodupkey; by bene_id; run;

```

Output 9.3 shows the first 10 observations of the output of Step 9.7.

Output 9.3: Print of Beneficiaries Who Received Diabetic Eye Exams

PRINT OF BENEFICIARIES WHO RECEIVED DIABETIC EYE EXAMS

Obs	BENE_ID	eye_svc
1	0004F0ABD505251D	1
2	00052705243EA128	1
3	00070B63745BE497	1
4	0007F12A492FD25D	1
5	0013E139F1F37264	1
6	00157F1570C74E09	1
7	001876C5D3F3F6AA	1
8	0018BD6F2F493452	1
9	00196F0702489342	1
10	001B371802E7EF58	1

Algorithms: Hospital Readmissions for Beneficiaries with COPD

Let's look for readmissions within 30 days of discharge for a hospitalization for any reason. As you advance your skills, you will be asked to program more complicated algorithms. For example, you may be asked to calculate readmission after a specific event, like a hospitalization for COPD or Acute Myocardial Infarction (AMI). In our case, we are simply looking for a readmission within 30 days of any hospitalization for our population of beneficiaries identified as having COPD, so we do not need to worry about identifying any diagnosis codes. In some cases, the reader may wish to use both prior and subsequent years of data in order to identify admissions and subsequent readmissions that occurred later in the study year, as well as early in the study year as a result of an admission in the year preceding the study year (in our case, calendar year 2010). For our simple instructional example, this will not be necessary.

First, in Step 9.8, we merge the file identifying beneficiaries with COPD created in Step 9.4 (CND.DIA_COPD_COND) with our inpatient claims data created in Chapter 7 (UTL.IP_2010_FNL_SS). We keep only those inpatient claims for beneficiaries with COPD.

```

/* STEP 9.8: JOIN INPATIENT CLAIMS WITH LIST OF BENEFICIARIES WITH COPD
*/
data copd_ip;
    merge cnd.dia_copd_cond(in=a where=(copd_flag=1))
    utl.ip_2010_fnl_ss;
    by bene_id;
    if a;
run;

```

Next in Step 9.9, we present code to look through the inpatient claims to identify admissions to a hospital that occurred within 30 days of discharge for a previous admission.¹⁰ Specifically, we use PROC SQL code to create a data set called CND.READM_COPD_COND that contains the beneficiary identifier (**BENE_ID**), discharge date (**DISCHRGDT**), and admission date (**ADMSN_DT**) on claims for beneficiaries with COPD representing readmissions to a hospital that occur 30 days after an initial admission. Note that we only select the **BENE_ID** from the data set with the alias “a” in order to avoid a warning in our log that the variable **BENE_ID** already exists on the output data set CND.READM_COPD_COND.

```

/* STEP 9.9: IDENTIFY READMISSIONS WITHIN 30 DAYS OF ANY DISCHARGE */
proc sql;
    create table cnd.readm_copd_cond as
    select a.*, b.readm
    from copd_ip(keep=bene_id dschrgdt) a, copd_ip(keep=bene_id
admsn_dt rename=(admsn_dt=readm)) b
    where a.bene_id=b.bene_id
    having ((readm gt dschrgdt+1) and (readm-dschrgdt+1<=30));
quit;

```

Output 9.4 shows the first 10 observations of the output of Step 9.9.

Output 9.4: Print of Hospital Readmissions for Beneficiaries with COPD

PRINT OF HOSPITAL READMISSIONS FOR BENEFICIARIES WITH COPD

Obs	BENE_ID	DSCHRGDT	readm
1	0007F12A492FD25D	20100612	20100616
2	007928DE5B0C4AA5	20100107	20100114
3	00F74CF706A0BE96	20100415	20100514
4	02A663B3BA049E9D	20100801	20100830
5	0388381BFCB2C4A3	20100122	20100124
6	05039EEC14848078	20100525	20100615
7	05039EEC14848078	20100619	20100707
8	09C052549BCFC935	20100208	20100213
9	1448403EBDF7E6B5	20100101	20100129
10	154379B03FD0C14B	20100614	20100620

Chapter Summary

In this chapter, we used our claims and enrollment data to study chronic conditions and quality outcomes. Specifically, we:

- Discussed the importance of studying chronic conditions. Many Medicare beneficiaries have chronic conditions; treating chronic conditions requires a high level of specialized and concerted medical care, and the treatment of chronic conditions is often looked at from the perspective of measuring quality outcomes.
- Programmed algorithms to identify beneficiaries with diabetes or COPD by defining the occurrence of certain diagnosis codes that identify these chronic conditions (including “rule out” diagnoses), searching the line items on claims for these diagnosis codes, and counting the occurrence of these codes by claim type.
- Programmed algorithms to identify evaluation and management services for beneficiaries with diabetes or COPD by merging beneficiaries with these conditions with the files we created in Chapter 7 to define E&M visits.

- Programmed simple algorithms to identify the occurrence of diabetic eye exams by searching for procedure codes that identify retinal eye exams in the Part B carrier data for beneficiaries identified as diabetic.
- Programmed simple algorithms to identify hospital readmissions for beneficiaries with COPD within 30 days of a discharge for any hospitalization.

Exercises

1. When using carrier claims to identify beneficiaries with diabetes, investigators often look for two claims separated by one or more days. In this case, we are looking for a diabetes diagnosis on two separate claims, but those claims could occur during the same time period. Can you modify Step 9.1 through Step 9.4 to identify and count diagnoses of diabetes on carrier claims occurring on separate days?
1. Can you modify the algorithms in Step 9.1 through Step 9.4 to identify beneficiaries with prostate cancer? Define beneficiaries with prostate cancer using the specifications from the CCW Medicare Administrative Data User Guide, available for download at http://www.ccwdata.org/cs/groups/public/documents/document/ccw_userguide.pdf.
2. Can you adapt the algorithms in Step 9.8 and Step 9.9 to look for readmissions within 30, 60, and 90 days of discharge for a hospitalization for AMI? Use the inpatient claims for our population of beneficiaries with COPD as your starting point.
3. Can you use the MBSF data received from CMS's data distribution contractor (i.e., our original enrollment data, not reduced to contain information only for those beneficiaries identified as continuously enrolled in Medicare FFS for all twelve months of calendar year 2010) to identify beneficiaries that died during the year, and subsequently examine their claims data to find evidence of diabetes or COPD? Compare the mortality rates of beneficiaries with diabetes with beneficiaries with COPD, as well as the overall population of all beneficiaries in the MBSF data.

¹ This definition is taken from the Missouri Department of Health and Senior Services, available at <http://health.mo.gov/living/families/schoolhealth/glossary.php>.

² For a full list of the chronic conditions identified in CMS's Chronic Condition Warehouse, see the Chronic Condition Reference List, available for download from the CCW website at http://www.ccwdata.org/cs/groups/public/documents/document/ccw_conditionreferencelist2011.pdf.

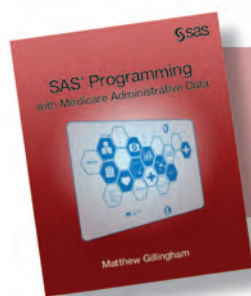
³ For this and more information on beneficiary counts by chronic condition algorithm, see the Comparison of Medicare Beneficiary Counts for Chronic Condition Algorithms, available on the CCW website at http://www.ccwdata.org/cs/groups/public/documents/document/ccw_web_table_b5.pdf.

⁴ This description of the CCW is taken from the CCW website, available at <http://www.ccwdata.org/index.htm>.

⁵ For instruction on how to identify diabetes, COPD, and other chronic conditions, see The Chronic Condition Data Warehouse Medicare Administrative Data User Guide, available for download from the CCW website at http://www.ccwdata.org/cs/groups/public/documents/document/ccw_userguide.pdf.

- ⁶ When identifying beneficiaries with diabetes using carrier claims, investigators often look for two claims separated by one or more days. In this case, we are looking for a diabetes diagnosis on two separate claims, but those claims could occur during the same time period. See the exercises section for a question on identifying carrier claims occurring on separate days.
- ⁷ Recall that the carrier file has 13 line items for each claim. In earlier chapters, we searched the 13 line-level procedure codes in carrier claims.
- ⁸ In Chapter 10, we will examine the overall rates of diabetes and COPD in our study population, as well as rates for the study population by state and county.
- ⁹ We use procedure codes to identify routine eye exams, but there can be more detailed definitions depending on the purpose of the investigation.
- ¹⁰ This algorithm is adapted from a discussion on The University of Georgia's listserv, available at <http://listserv.uga.edu/cgi-bin/wa?A2=ind0612b&L=sas-l&P=21025>. It is simple, clean, and perfect for our purposes. As you perform more complicated analyses (like looking for admissions for specific reasons), you will write more complex readmission algorithms.

From *SAS® Programming with Medicare Administrative Data* by Matthew Gillingham. Copyright © 2014, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



From *SAS® Programming with Medicare Administrative Data*.
Full book available for purchase [here](#).

Index

A

ADMSN_DT variable 89n3
Affordable Care Act (2010) 11
age group, displaying results by 62–63
AGE_CATS variable 62–63, 125–126
algorithms
 about 3, 4, 47–51
 creating final enrollment data for remainder of programming 66
 creating or retaining data elements for displaying results by certain characteristics 59–66
 delimiting claims by continuously enrolled beneficiary population 72–73
 diabetic eye exams for beneficiaries with diabetes 116–118
 Evaluation and Management visits for beneficiaries with diabetes or COPD 114–116
 hospital readmissions for beneficiaries with COPD 118–120
 identifying beneficiaries with diabetes or COPD 108–114
 identifying continuously enrolled FFS beneficiaries 55–59
 measuring emergency department utilization 80–81
 measuring E&M payments 95–98
 measuring E&M utilization 73–74
 measuring inpatient hospital payments 98–100
 measuring inpatient hospital utilization 75–79
 measuring outpatient visit information 84–85
 measuring total Part A payments 100–102
 measuring utilization of ambulance services 81–84
 measuring utilization of SNF, Home Health Agency, and hospice care 85–87

 presenting inpatient length of stay information by state 131–133
 presenting mean Medicare Part A payments per beneficiary by state and county 133–135
 presenting rates of diabetic eye exams 135–137
 presenting selected measurements of utilization, cost to Medicare, and quality outcomes 125–131
 transforming base claim and line level data sets into single claim-level files 47–51
ALS (Lou Gehrig's disease) 10
AMB_CARR_SVC variable 83–84
ambulance utilization, presenting 125–127
American Medical Association (AMA) 23, 89n1
A_MO_CNT variable 56
archiving materials for future use 138–139
'Are You a Hospital Inpatient of Outpatient' pamphlet 27n7

B

Balanced Budget Act (1997) 10
Bannister, Wade 89n8
base claim carrier data, transforming into claim-level files 48–51
base claim data sets, transforming into single claim-level files 47–48
beneficiaries
 with diabetes or COPD 108–116
 with end-stage renal disease (ESRD) 68n2
beneficiary identifier 22–23
BENE_HI_CVRAGE_TOT_MONS variable 56, 57
BENE_HMO_CVRAGE_TOT_MONS variable 56, 57
BENE_ID variable 22, 48, 49, 50, 51, 66, 71, 73, 78, 83, 86, 101–102, 116, 119, 125, 128
BENE_SMI_CVRAGE_TOT_MONS variable 56, 57

black lung disease 12–13

B_MO_CNT variable 56

C

CAHs (critical access hospitals) 75

carrier claims 122n6

carrier file 122n7

CCW (Chronic Conditions Data Warehouse)

68n3, 108, 121n2, 121n3, 121n4,
121n5

Centers for Medicare & Medicaid Service
(CMS) 7, 10, 41–42

chiropractic services 10

chronic conditions, identifying

about 105–106, 121n2, 121n5

diabetic eye exams for beneficiaries with
diabetes 116–118

Evaluation and Management (E&M) visits
for beneficiaries with Diabetes or
COPD 114–116

hospital readmissions for beneficiaries with
COPD 118–120

identifying beneficiaries with Diabetes or
COPD 108–114

peculiarities of Medicare data 107

programming plan 107–108

Chronic Conditions Data Warehouse (CCW)

68n3, 108, 121n2, 121n3, 121n4,
121n5

claim forms 16n11

claims data

See also data

basics of 70–72

commonly retained elements in 22–24

decrypting 46

inpatient 20–21

loading 46–47

receiving 45–46

requesting 40–41

sources of 18, 40–41

structure of 18

CLAIM_SEGMENT variable 86–87

CLAIM_SEGMENT_ID variable 86

CLM_ID variable 48, 49, 50, 51

CLM_LN variable 48, 49

CLM_LOOP macro 100, 109

CLM_LOOP2 macro 111

CMS (Centers for Medicare & Medicaid
Service) 7, 10, 41–42

CMS-1450 claim form 20, 21

CMS-1500 claim form 16n11, 20

CNTY_CD variable 64

Comparison of Medicare Beneficiary Counts for
Chronic Condition Algorithms
121n3

CONTENTNL_AB_2010 variable 57, 58, 59

CONTENTNL_HMO_2010 variable 57, 58, 59

COPD, beneficiaries with 108–116

COPD_CARR_SUM variable 111, 113

COPD_FLAG variable 113, 130

COPD_HH_SUM variable 111

COPD_IP_SUM variable 111

COPD_OP_SUM variable 111, 113

COPD_SN_SUM variable 111

costs, to Medicare

about 91–93

basics of 93–94

measuring Evaluation and Management
(E&M) payments 95–98

measuring inpatient hospital payments 98–
100

measuring total Part A payments 100–102

programming plan 93–94

coverage 11–13

coverage characteristics, displaying results by
59–61

coverage publications, Medicare 27n1

CPT (Current Procedural Terminology) codes
23

creating

data elements for displaying results 59–66

final enrollment data for remainder of
programming 66

critical access hospitals (CAHs) 75

Current Procedural Terminology (CPT) codes
23

D

data

See also claims data*See also* enrollment data

about 1–2, 17–18

algorithms 47–51

available from CMS 41–42

commonly retained elements in
administrative claims data 22–24

completing paperwork 43–45

contacting ResDAC 43–45

Durable Medical Equipment (DME) data
19–20

home health claims data 21

hospice claims data 22

identification of Emergency Department
(ED) utilization 25

identification of surgical services 26

inpatient claims data 20–21

master beneficiary summary file 24

outpatient claims data 20

Part B carrier claims data 19

peculiarities of 107

provider data 24–25

requirements and request specifications 42–
43Skilled Nursing Facility (SNF) claims data
21

data destruction 139

data dictionaries 68n3

data distribution contractor 18

Data Entrepreneurs' Synthetic Public Use File
(DE-SynPUF) 7, 27n4

Data Extract System (DESY) 18, 27n5, 40–41

data sets

loading claims and enrollment data into 46–
47

size and efficiency of 104n7

transforming 47–51

data storage, planning 35–36

Data Use Agreement (DUA) 8n6, 44, 138

date of death variable 57, 68n4

DEATH_2010 variable 57, 58, 59

DEATH_DT variable 57

decrypting claims data 46

delimiting claims by continuously enrolled
beneficiary population 72–73

DESY (Data Extract System) 18, 27n5, 40–41

diabetes

beneficiaries with 108–116

identifying 121n5

diabetic eye exams, for beneficiaries with
diabetes 116–118

DIA_FLAG variable 113, 116, 130

Diagnosis Present on Admission (POA) indicator
23

displaying results

by age group 62–63

by coverage characteristics 59–61

by geographic characteristics 63–66

by month of death 59–61

by race 59–61

by sex 59–61

DME (durable medical equipment) 19–20, 72

DME file 71

DSCHRG variable 89n3

DUA (Data Use Agreement) 8n6, 44, 138

durable medical equipment (DME) 19–20, 72

E

ED_SVC variable 80–81

electronic health records (EHRs) 2

eligibility 11, 15n2

E&M

See Evaluation and Management (E&M)

EM_COST variable 95

Emergency Department (ED) utilization
identification of 25

measuring 80–81

EM_SVC variable 74, 95, 115

EM_TOTBENE_PROV variable 128, 129

EM_TOTCOST_BENE variable 128

EM_TOTCOST_PROV variable 128

END2 variable 86

end-stage renal disease (ESRD) 10, 11, 68n2

enrolled FFS beneficiaries, identifying 55–59

enrollment, in Medicare 11
enrollment data
 about 53–54
 basics of 54–55
 creating final data for remainder of
 programming 66
 creating/retaining data elements for
 displaying results 59–66
 decrypting 45–47
 identifying enrolled FFS beneficiaries 55–
 59
 loading 45–47
 receiving 45–47
 requesting 40–41
 sources of 40–41
Evaluation and Management (E&M)
 measuring payments 95–98
 measuring utilization 73–74
 visits for beneficiaries with Diabetes or
 COPD 114–116
evaluative studies 8n4
examples
 identification of Emergency Department
 (ED) utilization 25
 identification of surgical services 26
 project 3–5
EYE_SVC variable 116, 135, 136

F

Federal Information Processing Standard (FIPs)
 coding system 68n7
Federal Insurance Contributions Act (FICA)
 taxes 10
FFS (Medicare Fee-for-Service) 11–12, 54
flowcharts
 about 38n2
 planning 33–35
FORMAT procedure 60, 62
framework, of this book 2–3
FREQ procedure 58, 61, 63, 77, 79, 115, 116,
 130, 132, 133, 136
FROM_DT variable 51, 85–87, 89n7

G

geographic characteristics, displaying results by
 63–66

H

HCPCS_CD1-HCPCS_CD13 variable 74
HCPCS-General Information (website) 28n14
Health Care Provider Taxonomy Code (HPTC)
 24–25
health maintenance organizations (HMOs) 12
"health reform legislation" 11
Healthcare Common Procedure Coding System
 (HCPCS) codes 23, 89n1
HHAs (Home Health Agencies) 21, 71, 104n5
HHRGs (home health resource groups) 21
HI (Hospital Insurance) 11
HIC variable 22, 28n13
HICAN variable 22
HMO coverage variable 56
HMOs (health maintenance organizations) 12
Home Health Agencies (HHAs) 21, 71, 104n5
home health claims data 21
home health resource groups (HHRGs) 21
hospice claims data 22
hospice claims file 71
hospice service 10
Hospital Compare 89n4
Hospital Insurance (HI) 11
hospital readmissions, for beneficiaries with
 COPD 118–120
Hospital Referral Regions (HRRs) 63, 68n6
Hospital Service Areas (HSAs) 63, 68n6
"How to Identify Emergency Room Services in
 the Medicare Claims Data" 89n5
HPTC (Health Care Provider Taxonomy Code)
 24–25

I

ICD-10 28n15
identifying
 See also chronic conditions, identifying
 beneficiaries with Diabetes or COPD 108–

Emergency Department (ED) utilization 25
 enrolled FFS beneficiaries 55–59
 IDR (Integrated Data Repository) 18, 27n5, 40–41
 Inpatient Prospective Payment System (IPPS) 75
 inpatient psychiatric care 12
 inpatients
 measuring hospital payments 98–100
 measuring hospitalization utilization 75–79
 presenting length of stay information by state 131–133
 Integrated Data Repository (IDR) 18, 27n5, 40–41
 International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM) 23
 International Classification of Diseases, Tenth Revision, Clinical Modification (ICD-10-CM) 28n15
 IP_ALOS variable 78, 132
 IP_LOS variable 76, 132
 IPPS (Medicare Inpatient Prospective Payment System) 75
 IP_TOTCOST_BENE variable 98

J

Johnson, Lyndon 10

L

LDS (Limited Data Sets) 42
 legislation
 Affordable Care Act (2010) 11
 Balanced Budget Act (1997) 10
 Medicare Prescription Drug, Improvement, and Modernization Act (MMA) (2003) 10
 libname CND 'C:DATA09_Conditions' 36
 libname CST 'C:DATA08_Cost' 36
 libname ENR 'C:DATA06_Enrollment' 36
 libname ETL 'C:DATA05_ETL' 35–36
 libname FNL 'C:DATA10_Results' 36
 libname SRC 'C:DATA00_Source_Data' 35

libname UTL 'C:DATA07_Utilization' 36
 LIBNAMES, planning 35–36
 Limited Data Sets (LDS) 42
 line level data sets, transforming into single claim-level files 47–48
 LINE_ICD_DGNS_CD variable 49
 line-level carrier data, transforming into claim-level files 48–51
 line-level procedure codes 122n7
 LINEPMT variables 95, 97
 loading claims data 46–47
 long-term care services 12
 long-term disabilities 10
 Lou Gehrig's disease (ALS) 10

M

MA (Medicare Advantage) 12
 MA prescription drug plans (MA-PDPs) 12
 Master Beneficiary Summary File (MBSF) 24, 28n16, 42, 55, 68n7
 MEANS procedure 134–135
 measuring
 See also costs, to Medicare
 See also utilization of services
 Emergency Department (ED) utilization 80–81
 Evaluation and Management (E&M) payments 95–98
 Evaluation and Management (E&M) utilization 73–74
 inpatient hospital payments 98–100
 inpatient hospitalization utilization 75–79
 outpatient visit information 84–85
 total Part A Payments 100–102
 utilization of ambulance services 81–84
 utilization of SNF, Home Health Agency, and Hospice Care 85–87
 mechanics, of Medicare 13–14
 Medicaid 15n5
 Medicare
 See also costs, to Medicare
 See also specific topics
 about 1, 2, 9–11

- beneficiary identifier 22–23
- eligibility 11
- enrollment 11
- mechanics of 13–14
- timeline for 15n3
- what is covered 11–12
- what is not covered 12–13
- Medicare Administrative Contractor (MAC) 13, 19
- Medicare Administrative data
 - See* claims data
 - See* data
 - See* enrollment data
- Medicare Advantage (MA) 12
- 'Medicare and Home Health Care' booklet 28n10
- Medicare and You 15n7
- Medicare Benefits 15n6
- Medicare claims data
 - See* claims data
- 'Medicare Coverage of Durable Medical Equipment and Other Devices' booklet 27n6
- 'Medicare Coverage of Skilled Nursing Facility Care' booklet 16n10, 27n9
- Medicare data
 - See* data
- Medicare enrollment data
 - See* enrollment data
- Medicare Fee-for-Service (FFS) 11–12, 54
- 'Medicare Hospice Benefits' booklet 28n11
- Medicare Inpatient Prospective Payment System (IPPS) 75
- Medicare Part A 11, 54, 56, 70
- Medicare Part B 11–12, 19, 54, 56, 70–71
- Medicare Part C 2, 12, 54, 55
- Medicare Part D 1, 12
- Medicare payments for Evaluation and Management services by provider, presenting 127–129
- Medicare Prescription Drug, Improvement, and Modernization Act (MMA) (2003) 10

- Medicare Secondary Payer 68n2
- Medicare Severity-Diagnosis Related Groups (MS-DRGs) 23, 75
- Medicare+Choice program 10
- medication 15n8
- MedPAC 104n1
- MedPAR file 27n8
- Metropolitan Statistical Area (MSA) 63
- Missouri Department of Health and Senior Services 121n1
- mlogic 104n6
- month of death, displaying results by 59–61
- mprint 104n6
- MSA (Metropolitan Statistical Area) 63
- MSABEA file 68n8
- MS-DRGs (Medicare Severity-Diagnosis Related Groups) 23, 75

N

- National Plan and Provider Enumeration System (NPPES) 24–25, 28n17
- National Provider Identifier (NPI) 23
- "Negative Payment Amounts in the Medicare Claims Data" 104n4
- nodupkey sort 68n9
- Non-Identifiable Files 42

O

- OP_VISIT_CNT variable 84
- "original fee-for-service (FFS)" 11–12, 54
- Outpatient Prospective Payment System (OPPS) 20
- outpatients
 - claims data 20
 - measuring visit information 84–85
 - prescription drugs 10
- output and project disposition
 - about 123–124
 - accomplishments 124
 - presenting inpatient length of stay information by state 131–133

- presenting mean Medicare Part A payments per beneficiary by state and county 133–135
 - presenting rates of diabetic eye exams 135–137
 - presenting selected measurements of utilization, cost to Medicare, and quality of outcomes 125–131
 - production execution in batch 138
 - programming plan 124–125
 - project disposition 138–139
- P**
- paperwork, completing 43–45
 - Part A 11, 54, 56, 70
 - Part B 11–12, 19, 54, 56, 70–71
 - Part C 2, 12, 54, 55
 - Part D 1, 12
 - PA_TOTCOST_BENE variable 134
 - payment standardization 94
 - PDPs (prescription drug plans) 12
 - physical screening 10
 - physical therapy 10
 - planning
 - about 29–30
 - data storage 35–36
 - flowchart 33–35
 - LIBNAMES 35–36
 - quality assurance (QA) plan 33
 - requirements 32
 - SDLC 30–32
 - technical specifications 36–37
 - PMT_AMT variable 76, 98, 100, 104n3, 109
 - POA (Diagnosis Present on Admission) indicator 23
 - PPS (Prospective Payment System) 2, 10, 14, 16n3, 92
 - preferred provider organizations (PPOs) 12
 - prescription drug plans (PDPs) 12
 - presenting
 - ambulance utilization 125–127
 - inpatient length of stay information by state 131–133
 - mean Medicare Part A payments per beneficiary by state and county 133–135
 - Medicare payments for Evaluation and Management services by provider 127–129
 - rates of diabetes and COPD 129–131
 - rates of diabetic eye exams 135–137
 - selected measurements of utilization, cost to Medicare, and quality of outcomes 125–131
 - preventive screening 10
 - PRFNPI variable 97
 - PRINT procedure 84, 129
 - Privacy Act (1974) 44
 - procedure codes 122n9
 - procedures
 - FORMAT 60, 62
 - FREQ 58, 61, 63, 77, 79, 115, 116, 130, 132, 133, 136
 - MEANS 134–135
 - PRINT 84, 129
 - SQL 52n5, 78, 84, 93, 95–97, 100, 119, 128
 - SUMMARY 126–127
 - production execution in batch 138
 - programming, in SAS to define continuous enrollment in Medicare FSS 57–59
 - project disposition
 - See* output and project disposition
 - Prospective Payment System (PPS) 2, 10, 14, 16n13, 92
 - provider data 24–25
 - provider identifier 23
 - PROVIDER variable 75–76
 - PROVTYPE variable 76
 - Public Use Files (PUFs) 42
- Q**
- quality assurance (QA) procedures 30, 33
- R**
- race, displaying results by 59–61
 - RACE variable 59–61

rates of diabetes and COPD, presenting 129–131
 rates of diabetic eye exams, presenting 135–137
 receiving claims data 45–46
 requesting data 40–41
 requirements, for planning 32
 Research Data Assistance Center (ResDAC)
 16n12, 18, 28n18, 40–41, 43–45,
 52n2, 68n4, 89n5, 89n7, 104n3
 Research Data Distribution Center 18
 Research Identifiable Files (RIFs) 42
 retaining data elements for displaying results
 59–66
 risk adjustment 94

S

SDLC (Systems Development Life Cycle) 30–
 32
 services, codes identifying 23
 services, utilization of
 about 69–70
 basics of Medicare claims data 70–72
 delimiting claims by continuously enrolled
 beneficiary population 72–73
 measuring 81–84
 measuring emergency department utilization
 80–81
 measuring E&M utilization 73–74
 measuring inpatient hospitalization
 utilization 75–79
 measuring outpatient visit information 84–
 85
 measuring utilization of ambulance services
 81–84
 measuring utilization of SNF, Home Health
 Agency, and Hospice Care 85–87
 programming plan 72
 sex, displaying results by 59–61
 SEX variable 59–61
 Shannon, Doug 89n8
 skilled nursing facilities (SNFs) 11, 21
 SMI (Supplemental Medical Insurance) 11–12
 SNF file 71
 Social Security Disability Insurance (SSDI) 11

source data 18, 27n3, 40–41, 52n1, 52n3
 specifying programming for continuous
 enrollment in Medicare FFS 56
 speech therapy 10
 SQL procedure 52n5, 78, 84, 93, 95–97, 100,
 119, 128
 SSA variable 64, 65
 SSDI (Social Security Disability Insurance) 11
 standardization, payment 94
 STATE_CD variable 64
 STAY_CNT variable 78
 STUDY_AGE variable 62, 68n5, 125–126
 substring 89n2
 SUMMARY procedure 126–127
 Supplemental Medical Insurance (SMI) 11–12
 support, online, for this book 7
 surgical services, identification of 26
 Synthetic Public Use File (DE-SynPUF) 7, 27n4
 Systems Development Life Cycle (SDLC) 30–
 32

T

technical specifications, planning 36–37
 terminally ill individuals 10
 THRU_DT variable 51, 51f, 85–87, 89n7
 TOT_AMB variable 83–84
 TOT_AMB_SVC variable 125–126
 TOT_IP_LOS variable 78
 transforming base claim and line level data sets
 47–51
 Truman, Harry 10

U

UB-92 form 20
 University of Georgia listserv 122n10
 UNIX 138
 utilization of services
 about 69–70
 basics of Medicare claims data 70–72
 delimiting claims by continuously enrolled
 beneficiary population 72–73
 measuring 81–84

measuring emergency department utilization 80–81
 measuring E&M utilization 73–74
 measuring inpatient hospitalization utilization 75–79
 measuring outpatient visit information 84–85
 measuring utilization of ambulance services 81–84
 measuring utilization of SNF, Home Health Agency, and Hospice Care 85–87
 programming plan 72
 utilization of SNF, Home Health Agency, and Hospice Care, measuring 85–87

V

VALSUG 89n8

variables

about 28n12
 ADMSN_DT 89n3
 AGE_CATS 62–63, 125–126
 AMB_CARR_SVC 83–84
 A_MO_CNT 56
 BENE_HI_CVRAGE_TOT_MONS 56, 57
 BENE_HMO_CVRAGE_TOT_MONS 56, 57
 BENE_ID 22, 48, 49, 50, 51, 66, 71, 73, 78, 83, 86, 101–102, 116, 119, 125, 128
 BENE_SMI_CVRAGE_TOT_MONS 56, 57
 B_MO_CNT 56
 CLAIM_SEGMENT 86–87
 CLAIM_SEGMENT_ID 86
 CLM_ID 48, 49, 50, 51
 CLM_LN 48, 49
 CNTY_CD 64
 CONTENRL_AB_2010 57, 58, 59
 CONTENRL_HMO_2010 57, 58, 59
 COPD_CARR_SUM 111, 113
 COPD_FLAG 113, 130
 COPD_HH_SUM 111
 COPD_IP_SUM 111
 COPD_OP_SUM 111, 113

COPD_SN_SUM 111
 date of death 57, 68n4
 DEATH_2010 57, 58, 59
 DEATH_DT 57
 DIA_FLAG 113, 116, 130
 DSCHRG 89n3
 ED_SVC 80–81
 EM_COST 95
 EM_SVC 74, 95, 115
 EM_TOTBENE_PROV 128, 129
 EM_TOTCOST_BENE 128
 EM_TOTCOST_PROV 128
 END2 86
 enrollment 56
 EYE_SVC 116, 135, 136
 FROM_DT 51, 85–87, 89n7
 HCPCS_CD1-HCPCS_CD13 74
 HIC 22, 28n13
 HICAN 22
 HMO coverage 56
 IP_ALOS 78, 132
 IP_LOS 76, 132
 IP_TOTCOST_BENE 98
 LINE_ICD_DGNS_CD 49
 LINEPMT 95, 97
 OP_VISIT_CNT 84
 PA_TOTCOST_BENE 134
 PMT_AMT 76, 98, 100, 104n3, 109
 PRFNPI 97
 PROVIDER 75–76
 PROVTYPE 76
 RACE 59–61
 SEX 59–61
 SSA 64, 65
 STATE_CD 64
 STAY_CNT 78
 STUDY_AGE 62, 68n5, 125–126
 THRU_DT 51, 85–87, 89n7
 TOT_AMB 83–84
 TOT_AMB_SVC 125–126
 TOT_IP_LOS 78
 V_DOD_SW 57
 V_DOD_SW variable 57

Virtual Research Data Center (VRDC) system
18, 40–41
voluntary prescription drug coverage 12

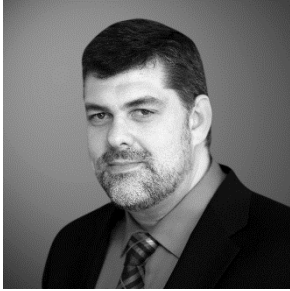
W

War on Poverty 10

Numerics

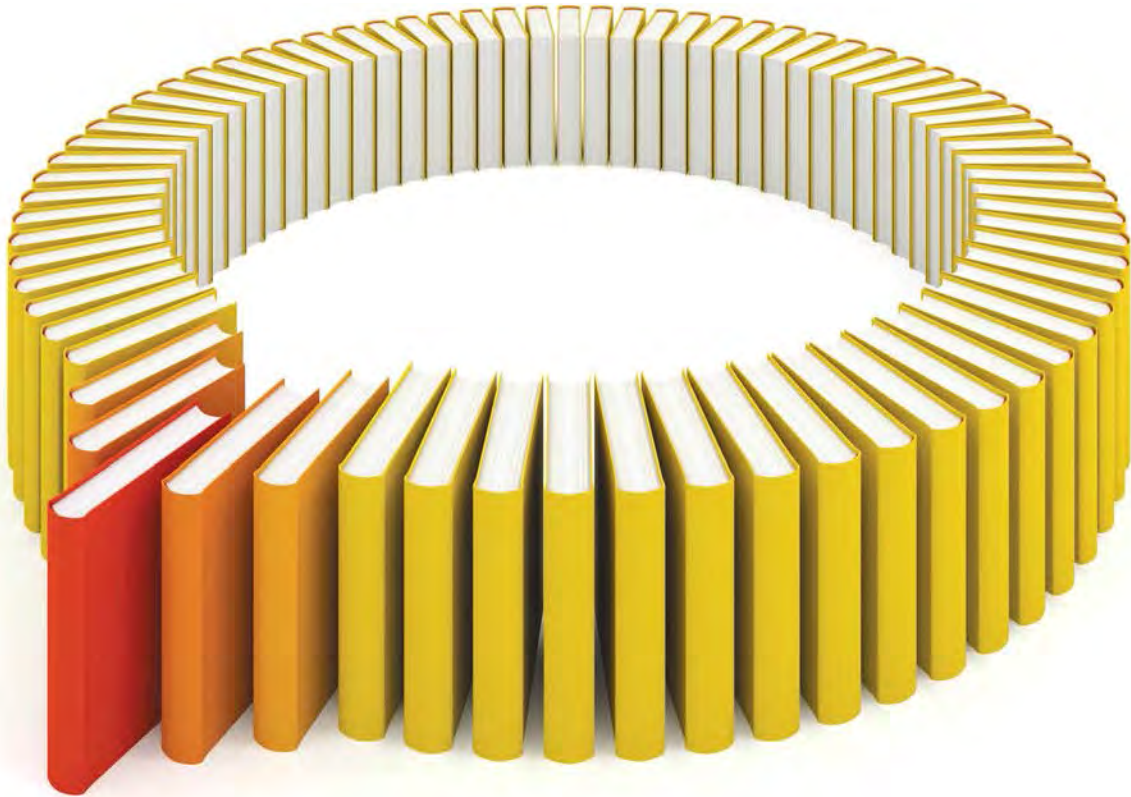
05_ETL.sas program 34, 35
06_Enrollment.sas program 35
07_Utilization.sas program 35
08_Cost.sas program 35
09_Conditions.sas program 35
10_Presenting_Output.sas program 35
837i electronic format 21

About The Author



Matthew Gillingham is the director of health research systems at Mathematica Policy Research. His career in health care data analytics began with analyzing commercial claims and enrollment data. For more than ten years, he has used SAS to build software that analyzes administrative health care and Medicare administrative data. Specifically, Gillingham directs systems research and development work that uses large Medicare claims and enrollment data files to monitor and report on health care quality and utilization for the Centers for Medicare & Medicaid Services. In addition, he provides technical assistance and quality assurance on projects that use Medicare data to create analytic files for program evaluation, monitoring, and other research purposes.

Learn more about this author by visiting his author page at <http://support.sas.com/publishing/authors/gillingham.html>. There, you can download free chapters, access example code and data, read the latest reviews, get updates, and more.



Gain Greater Insight into Your SAS[®] Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

 support.sas.com/bookstore
for additional books and resources.


THE POWER TO KNOW.®