

'PEDIATRIC MEDICAL COMPLEXITY ALGORITHM v1.0.sas'

Programmer(s): Dorothy Lyons, Peter Woodcox
First Steps Database
Research and Data Analysis Division
Washington State Department of Social and Health Services

Date: September 2012, final revision for v1.0 December 2012

Description:

This program implements the Pediatric Medical Complexity Algorithm to identify children with complex and non-complex chronic conditions using Medicaid claims data and to distinguish them from children with neither chronic nor chronic complex conditions (healthy children). 'Complex' and 'Non-Complex' designations are assigned based on whether a child's condition(s) identified by ICD-9 code(s) can be considered chronic, malignant, or progressive, and whether multiple body systems are involved.

The AHRQ-funded Center of Excellence on Quality of Care Measures for Children with Complex Needs developed consensus definitions for children with complex chronic disease (C-CD), children with non-complex chronic disease (NC-CD), and children without CD (healthy). Development and testing of this PMCA was supported by the COE (Rita Mangione-Smith, MD MPH, Principal Investigator).

To assign involved body system(s) and progressive status, Seattle Children's Research Institute evaluated conditions defined by diagnostic codes outlined in the Chronic Illness and Disability Payment System (CDPS) version 5.3, and identified conditions of interest based on their chronic nature.

References:

Chronic Illness and Disability Payment System (CDPS), University of California, San Diego.

Input Source Data:

This program was designed for use with three years of data from a Medicaid claims file that includes all claims (inpatient, outpatient, managed care encounter data) and has at least the following fields:

- 1) Person ID unique to a person
- 2) Claim ID unique to a claim
- 3) ICD-9 diagnosis code(s)

In the source data for which this program was designed, a single claim may have multiple lines,

and a single record/line contains 25 fields for icd-9 codes.

The program rolls up assignments by claim and sets an array for icd-9 codes, so that it accommodates source files that have a different number of possible diagnosis code fields, or source files that have one row for each diagnosis code.

Format for ICD-9 codes:

This program assumes that incoming icd-9 codes (as the variable called 'icdvar') are in xxx.xx character format with leading zeros for codes that do not have 3 digits before the decimal. The program then strips out the decimals. If original source data do not have decimals, the stripping command can be deleted from the processing.

The program is designed to process records with 25 diagnosis code fields.

A macro statement has been provided at the start of the code to set the number of fields.

To change the number of diagnosis fields processed, change 'icdnum' in the SET VALUES FOR VARIABLES section to the number of fields in a single record of the source data.

The Process:

- 1) 'Claims' dataset: claims are read in and the decimals are stripped from the ICD-9 codes. If incoming data do not have decimals, the do-end 'compress' segment can be commented out. ICD-9 codes with or without decimals are thus accommodated.
- 2) Claims are sorted by person ID and claim ID.
- 3) 'Flagclaims' processing steps through the multiple lines (or single line) of individual claims for a person, identifying body systems involved, progressive status of a condition, and malignancy. The final result is a dataset where identifications of body system, progressivity, and malignancy have been rolled up to a single record per claim, with a single indication of any specified occurrence (i.e. once per claim).
- 4) 'Results' processing rolls up to one record per child, with
 - a) a single indication whether each body type is identified as affected,
 - b) a sum across claims for each body type affected
(i.e. how many claims for this child have this kind of indication?),
 - c) indication if a progressive condition, and
 - d) indication if malignancy.

Finally, this collected information is used to calculate the child's status.

Final Condition Definitions:

This program calculates two separate variables containing condition assignments for

'Complex Chronic', 'Non-complex Chronic', and 'Non-Chronic'.

The variable 'cond_less' contains values from the less conservative algorithm, and is designed to be used with less detailed data sources, such as those without outpatient claims.

The variable 'cond_more' contains values from the more conservative algorithm, and is designed to be used with more detailed data sources such as those including inpatient and outpatient claims.

Values for both of the above variables are

- '3 Complex Chronic'
- '2 Non-complex Chronic'
- '1 Non-Chronic'

Definitions of the categories assigned by the Algorithm:

The less conservative version (cond_less) calculates values as

- 'Complex Chronic':
 - 1) more than one body system is involved, or
 - 2) one or more conditions are progressive, or
 - 3) one or more conditions are malignant
- 'Non-complex Chronic':
 - 1) only one body system is involved, and
 - 2) the condition is not progressive or malignant
- 'Non-Chronic':
 - 1) no body system indicators are present, and
 - 2) the condition is not progressive or malignant

The more conservative version (cond_more) calculates values as

- 'Complex Chronic':
 - 1) more than one body system is involved,
and each must be indicated in more than one claim, or
 - 2) one or more conditions are progressive, or
 - 3) one or more conditions are malignant
- 'Non-complex Chronic':
 - 1) only one body system is indicated in more than one claim, and
 - 2) the condition is not progressive or malignant
- 'Non-Chronic':
 - 1) no body system indicators are present in more than one claim, and
 - 2) the condition is not progressive or malignant

Body Systems of interest and the variables used to indicate them:

| | |
|-----------------------|---------|
| cardiac | cardiac |
| craniofacial | cranio |
| dermatological | derm |
| endocrinological | endo |
| gastrointestinal | gastro |
| genetic | genetic |
| genitourinary | genito |
| hematological | hemato |
| immunological | immuno |
| malignancy | malign |
| mental health | mh |
| metabolic | metab |
| musculoskeletal | musculo |
| neurological | neuro |
| pulmonary-respiratory | pulresp |
| renal | renal |
| ophthalmological | opthal |
| otologic | otol |
| otolaryngological | otolar |

Datasets Created:

The FLAGCLAIMS dataset contains 1 record for each claim record, with the condition flags added.

The RESULTS dataset creates 1 record per client with accumulated indicators, and keeps final condition determinations.

Macro statements have been provided to more easily adapt the program to different data sources (see directions below).

Created December 2012. Future revisions to the ICD-9 classification system may result in the need for revisions to this program.

Inputs:

INDATA.CLAIMS (Source file with claims data)

Outputs:

OPUT.RESULTS_PMCA_v1_0 (Output file with 1 record per person and condition classifications)

DIRECTIONS:

- 1) SET INCOMING DATA LOCATION
- 2) SET NUMBER OF ICD-9 CODE FIELDS (i.e. 9 for dx1-dx9)
- 3) SET NAME OF INCOMING CLAIMS SOURCE DATASET
- 4) SET NAME OF UNIQUE PERSON ID
- 5) SET NAME OF VARIABLES CONTAINING ICD-9 CODES
- 6) SET NAME OF UNIQUE CLAIM CODE
- 7) SET OUTPUT LOCATION

*/

```

-----
SET VALUES FOR VARIABLES ETC
-----;

```

```

libname indata "C:\datafolder";          * !!! <----- 1) SET LOCATION OF CLAIMS DATA TO BE USED;

%let icdnum=25;                          * !!! <----- 2) SET NUMBER OF ICD9 FIELDS IN YOUR DATA;
%let sdata=claimsfile;                  * !!! <----- 3) SET NAME OF SOURCE DATASET;
%let sid=p1id;                          * !!! <----- 4) SET NAME OF SOURCE UNIQUE PERSON ID;
%let icdvar=dx;                          * !!! <----- 5) SET NAME OF VARIABLES CONTAINING ICD-9 CODES
                                         Assumes that multiple fields have the same root
                                         name plus numbers 1-? i.e. dx1 to dx25);

%let claimid=tcn;                        * !!! <----- 6) SET NAME OF UNIQUE CLAIM CODE;
libname oput "C:\outputfolder";          * !!! <----- 7) SET OUTPUT LOCATION FOR FINAL PROCESSED DATA;

```

```

-----
PROCESS DATA
-----;

```

*Read in the claims data. Keep client ID, date of service, and diagnosis codes, and strip out the ICD-9 decimal. If incoming ICD-9 data do not have decimals, the processing within this data step can be commented out;

```

data claims(keep=&sid &claimid &icdvar.1-&icdvar.&icdnum);
  set indata.&sdata;
  array &icdvar(&icdnum) &icdvar.1-&icdvar.&icdnum;
  do i = 1 to &icdnum;
    &icdvar{i} = compress(&icdvar{i},".");
  end;
run;

```

/* Sort records by client id and claim id in order to roll up designations by claim. */

```

proc sort data=claims out=claims2; by &sid &claimid; run;

/* Assign values to the appropriate flags based on conditions found and roll up by claim. */

data flagclaims(keep=&sid &claimid
               cardiac cranio derm endo gastro genito hemato immuno malign mh metab musculo neuro genetic opthal
               otol otolar pulresp renal progressive);
set claims2;
by &sid &claimid;

retain cardiac cranio derm endo gastro genetic genito hemato immuno malign mh
       metab musculo neuro opthal otol otolar pulresp renal progressive;

* preset all the flags to 0 for rollup to one record with designations per claim;

if first.&claimid then
  do;
    cardiac = 0; cranio = 0; derm = 0; endo = 0; gastro = 0; genetic = 0; genito = 0; hemato = 0;
    immuno = 0; malign = 0; mh = 0; metab = 0; musculo = 0; neuro = 0; opthal = 0;
    otol = 0; otolar = 0; pulresp = 0; renal = 0;
    progressive = 0;
  end;

array &icdvar(&icdnum) &icdvar.1-&icdvar.&icdnum;

do i = 1 to &icdnum;
  if &icdvar{i} in: ('010' '011' '012' '013' '014' '015' '016' '017' '018') then pulresp=1;

  if &icdvar{i} =:'030' then neuro=1;

  if &icdvar{i} =:'0402' then gastro=1;

  if &icdvar{i} in: ('042' '043' '044') then do; immuno=1; progressive=1; end;

  if &icdvar{i} in: ('046' '0582' '0785') then do; neuro=1; progressive=1; end;

  if &icdvar(i) =:'07953' then do; immuno=1; progressive=1; end;

  if &icdvar{i} in: ('090' '094' '095') then do; neuro=1; progressive=1; end;

  if &icdvar{i} =:'135' then immuno=1;

```

```
if &icdvar{i} =:'1363' then pulresp=1;

if &icdvar{i} =:'137' then do; pulresp=1; progressive=1; end;

if &icdvar{i} in: ('138' '139') then do; neuro=1; progressive=1; end;

if &icdvar{i} in: ('14' '15' '16' '17' '18' '19' '200' '201' '202' '203' '204' '205'
'206' '207' '208' '2090' '2091' '2092' '2093' '2097' '23877') then malign=1;

if &icdvar{i} in: ('240' '241' '242' '243' '244' '245'
'246' '249' '250' '2510' '252'
'2530' '2531' '2533' '2534' '2535' '2537' '2538'
'254' '255' '256' '257' '258') then endo=1;

if &icdvar{i}=:'2532' then do; endo=1; progressive=1; end;

if &icdvar{i} in: ('260' '261' '262'
'2630' '2632'
'264' '265' '266' '267'
'2680' '2681' '2682'
'273' '274') then metab=1;

if &icdvar{i} in: ('270' '271' '272' '2750') then do; metab=1; progressive=1; end;

if &icdvar{i} =:'2770' then do; pulresp=1; progressive=1; end;

if &icdvar{i} in: ('2771' '2774' '2777') then metab=1;

if &icdvar{i} in: ('2772' '2773' '2775' '27781' '27782' '27783'
'27784' '27785' '27786' '27787' '27788')
then do; metab=1; progressive=1; end;

if &icdvar{i} =:'2776' then immuno=1;

if &icdvar{i} =:'279' then do; immuno=1; progressive=1; end;

if &icdvar{i} =:'2810' then hemato=1;

if (&icdvar{i} =:'282' and &icdvar{i} ^=:'2825') then do; hemato=1; if &icdvar{i} in: ('2824' '2826') then progressive=1;

if &icdvar{i} =:'283' then hemato=1;

if &icdvar{i} =:'284' then do; hemato=1; if &icdvar{i} ^=:'2841' then progressive=1; end;
```

```
if &icdvar{i} in:('2850' '28521' '28522' '28529' '2858') then hemato=1;

if &icdvar{i} =:'286' then
do;
hemato=1;
if &icdvar{i} in: ('2860' '2861' '2862' '2863') then progressive=1;
end;

if &icdvar{i} in:('2871' '2873') then do; hemato=1; end;

if &icdvar{i} in:('28802' '2885') then immuno=1;
if &icdvar{i} in:('28801' '2881' '2882' '2884') then do; immuno=1; progressive=1; end;

if &icdvar{i} in:('28951' '28952' '28953'
'28981' '28983' '28989') then do; immuno=1; end;

if &icdvar{i} in:('2911' '2921' '2940') then mh=1;
if &icdvar{i} in:('2950' '2951' '2952' '2953' '2954' '2955' '2956' '2957' '2958') then do; mh=1; progressive=1; end;

if &icdvar{i} in: ( '296' '2971' '2973'
'2990' '2991' '2998'
'3001' '3003' '3007' '30081'
'3010' '3011' '3012' '3013' '3014' '3015' '3016' '3017' '3018'
'3039'
'3040' '3041' '3042' '3044' '3045' '3046' '3047' '3048' '3049'
'30722' '30723' '3073' '3077') then mh=1;

if &icdvar{i} in:('3071' '30751' ) then do; mh=1; progressive=1; end;

if &icdvar{i} in: ( '3100' '3101' '311'
'3120' '3121' '3122' '3123' '3124' '3125' '3126' '3127' '31281' '31282'
'31381' '3140' '3141' '3142'
'31501' '31502' '3151' '3152' '31531' '31532' '31534' '3154' '3155'
'317' '3180' '3181' '3182' '319') then mh=1;

if &icdvar{i} =:'326' then do; neuro=1; progressive=1; end;
if &icdvar{i} in:('32721' '32723' '32725') then neuro=1;

if &icdvar{i} =:'330' then do; neuro=1; progressive=1; end;

if &icdvar{i} in:('3313' '3314') then neuro=1;
```



```
if &icdvar{i} in:('3330' '3332' '3334' '3335' '3336''33371' '33391') then do; neuro=1; progressive=1; end;

if &icdvar{i} =:'334' then do; neuro=1; progressive=1; end;

if &icdvar{i} in:('335' '336') then do; neuro=1; progressive=1; end;

if &icdvar{i} =: '337' then neuro=1;

if &icdvar{i} in:('340' '341') then do; neuro=1; progressive=1; end;

if &icdvar{i} =:'342' then neuro=1;

if &icdvar{i} =:'343' then
do;
neuro=1;
if &icdvar{i} =:'3432' then progressive=1;
end;

if &icdvar{i} =:'344' then
do;
neuro=1;
if &icdvar{i} =:'3440' then progressive=1;
end;

if &icdvar{i} in:('345' '347') then neuro=1;

if &icdvar{i} in:('3481' '3482' '34881') then
do;
neuro=1;
if &icdvar{i} =:'3481' then progressive=1;
end;

if &icdvar{i} in:('350' '351' '352' '353' '354' '355' '356' '357' '358') then neuro=1;

if &icdvar{i} in:('3590' '3591' '3592' '3593' '3594' '3595' '3598') then musculo=1;

if &icdvar{i} in:('3602' '3603' '3604' '361'
'3620' '3621' '36226' '36227'
'36230' '36231' '36232' '36233' '36234' '36235' '36236' '36237'
'36240' '36241' '36242' '36243'
'36250' '36251' '36252' '36253' '36254' '36255' '36256' '36257')
```

```

'36260' '36261' '36262' '36263' '36264' '36265' '36266'
'36270' '36271' '36272' '36273' '36274' '36275' '36276' '36277'
'36285'
'363'
'3641' '3642' '3645' '3647' '3648'
'365'
'3690' '3691' '3692' '3693' '3694'
'36960' '36961' '36962' '36963' '36964' '36965' '36966' '36967' '36968' '36969'
'36970' '36971' '36972' '36973' '36974' '36975' '36976'
'377'
'3780' '3781' '3782' '3783' '3784' '3785' '3786' '3787' '3788'
'3790' '3791' '3792' '3793' '3794' '3795' '3796') then opthal=1;

if &icdvar{i} in:('385' '386' '387'
'3880' '3881' '3883' '3885') then otol=1;

if &icdvar{i} in:('3891' '3892' '3897') then neuro=1;
if &icdvar{i} =:'390' then immuno=1;

if &icdvar{i} in:('393' '394' '395' '396' '397' '398' '401' '402') then
do;
cardiac=1;
if &icdvar{i} in:('40201' '40211' '40291') then progressive=1;
end;

if &icdvar{i} =:'403' then
do;
renal=1;
if &icdvar{i} in:('40301' '40311' '40391') then progressive=1;
end;

if &icdvar{i} =:'404' then
do;
renal=1;
if &icdvar{i} in:('40401' '40411' '40491'
'40402' '40403' '40412' '40413' '40492' '40493') then progressive=1;
end;

if &icdvar{i} =:'405' then cardiac=1;

if &icdvar{i} in:('410' '411' '412') then do; cardiac=1; progressive=1; end;
if &icdvar{i} =:'414' and &icdvar{i} ^=:'4144' then do; cardiac=1; progressive=1; end;
if &icdvar{i} in:('416' '417') then do; cardiac=1; progressive=1; end;

```

```
if &icdvar{i} in:('424' '426') then cardiac=1;

if (&icdvar{i} =:'425' and &icdvar{i} ^=:'4259') then do; cardiac=1; progressive=1; end;

if &icdvar{i} in:('4270' '4271' '4273' '4274' '42781') then cardiac=1;

if (&icdvar{i} in:('428' '4291') and &icdvar{i} ^=:'4289') then do; cardiac=1; progressive=1; end;

if &icdvar{i} in:('433'
                '4372' '4373' '4374' '4375' '4376' '4377'
                '438') then do; neuro=1; progressive=1; end;

if &icdvar{i} in:('441') then do; cardiac=1; progressive=1; end;

if &icdvar{i} in:('442' '443') then cardiac=1;

if &icdvar{i} =:'446' then
do;
    immuno=1;

    if &icdvar{i} in:('4460' '4462' '4463') then
do;
    progressive=1;
end;
end;

if &icdvar{i} in:('447') then cardiac=1;

if &icdvar{i} =:'452' then do; cardiac=1; progressive=1; end;

if &icdvar{i} in:('4530' '45350' '45351' '45352'
                '45371' '45372' '45373' '45374' '45375' '45376' '45377' '45379'
                '4570' '4571' '4572') then cardiac=1;

if &icdvar{i} in:('4760' '4761') then pulresp=1;

if &icdvar{i} =:'491' then pulresp=1;

if &icdvar{i} =:'492' then do; pulresp=1; progressive=1; end;

if &icdvar{i} =:'493' then pulresp=1;
```

```
if &icdvar{i} in:('4940' '4941')      then do;  pulresp=1; progressive=1; end;

if &icdvar{i} =:'495'      then pulresp=1;

if &icdvar{i} =:'496'      then do;  pulresp=1; progressive=1; end;

if &icdvar{i} in:('501' '502' '503' '504' '505')      then pulresp=1;

if &icdvar{i} in:('515' '516') then do;  pulresp=1; progressive=1; end;

if &icdvar{i} in:('5190' '5193' '5194') then pulresp=1;

if &icdvar{i} =:'526'      then musculo=1;

if &icdvar{i} in: ('5270' '5271' '5277') then gastro=1;

if &icdvar{i} in:('5300' '53013' '5303' '5305' '5306' '53083' '53084' '53085') then gastro=1;

if &icdvar{i} in: ('531' '532' '533' '534') then gastro=1;

if &icdvar{i} in:('555' '556' '562' '5651'
                 '5690' '5691' '5692' '56944'
                 '56981' '56984' '56986' '56987')      then gastro=1;

if &icdvar{i} =:'571'      then do;  gastro=1; progressive=1; end;

if &icdvar{i} in:('5723' '5724' '5730') then do;  gastro=1; progressive=1; end;

if &icdvar{i} in:('5732' '5734' '5738' '5739'
                 '57511' '5755' '5756' '5758'
                 '5760' '5761' '5764' '5765' '5768'
                 '5772' '5778'
                 '5790' '5791' '5792' '5794') then gastro=1;

if &icdvar{i} =:'5771' then do;  gastro=1;progressive=1; end;

if &icdvar{i} =:'581'      then renal=1;

if &icdvar{i} in:('582' '583' '585' '586')      then do;  renal=1; progressive=1; end;

if &icdvar{i} =:'5880' then do;  musculo=1; progressive=1; end;

if &icdvar{i} in:('5881'
```

```

'59371' '59372' '59373' '59382'
'5960' '5961' '5962' '5963' '5964'
'59651' '59652' '59653' '59654' '59655'
'598'
'5991' '5992' '5993' '5994' '5995'
'59981' '59982' '59983') then renal=1;

if &icdvar{i} in:(
'60785'
'6083'
'617' '618' '619'
'6221'
'6230'
'6240'
'62920' '62921' '62922' '62923' '62929') then genito=1;

if &icdvar{i} =:'694' then derm=1;

if &icdvar{i} =:'6954' then do; immuno=1; progressive=1; end;

if &icdvar{i} in:('7010' '7018' '702' '7050' '707') then derm=1;

if &icdvar{i} =:'710' then
do;
immuno=1;
if &icdvar{i} in:('7100' '7108' '7109') then
do;
progressive=1;
end;
end;

if &icdvar{i} in:('712' '714') then immuno=1;

if &icdvar{i} in:('717'
'7180' '7182' '7183' '7184' '7185' '7186' '7187'
'7220' '7221' '7222' '7223' '7224' '7225' '7226' '7227' '7228') then musculo=1;

if &icdvar{i} in:('720' '721' '725') then immuno=1;

if &icdvar{i} in:('7281' '7282' '7286' '7287') then musculo=1;
if &icdvar{i} =:'7283' then do; musculo=1; progressive=1; end;

if &icdvar{i} in:('7301'

```

```

'7310' '7311' '7312' '7313' '7318'
'7320' '7321'
'7330' '7333' '7334' '7337' ) then musculo=1;

if &icdvar{i} in:( '73605' '73606' '73607' '73631' '73632'
'73671' '73672' '73673' '73674' '73675'
'73681') then musculo=1;

if &icdvar{i} in:( '7370' '7371' '7373' '7378' '7379'
'7384' '7385' '7386' '7387' ) then musculo=1;

if &icdvar{i} in:( '7400' '7401' '7402' '741') then do; neuro=1; progressive=1; end;

if &icdvar{i} =:'742' then
do;
neuro=1;
if (&icdvar{i} ^=:'7423') then progressive=1;
end;

if &icdvar{i} in:( '7430' '7431' '7432' '7434' '7435'
'74361' '74362' '74363' '74366' '74369') then do; ophthal=1; end;

if &icdvar{i} in:( '7440' '7442' '7443' '7444' '7449') then do; otolar=1; end;

if &icdvar{i} in:( '7450' '7451' '7452' '7453' '7456' '7457' '7458' '7459') then
do;
cardiac=1;
progressive=1;
end;

if &icdvar{i} =:'7454' then cardiac=1;

if (&icdvar{i} =:'746' and &icdvar{i} ^=:'7469') then
do;
cardiac=1;
if &icdvar{i} in:( '7462' '7467') then progressive=1;
end;

if &icdvar{i} =:'7474' then do; cardiac=1;progressive=1; end;

if &icdvar{i} in:( '7471' '74721' '74722' '74729' '7473' '74781' '74783' '74789') then cardiac=1;

if &icdvar{i} =:'748' then

```

```

        do,
            pulresp=1;
            if &icdvar{i} in:('7484' '7485' '7486') then progressive=1;
        end;

if &icdvar{i} =:'749' then cranio=1;

if &icdvar{i} in:('7501' '7502' '7503' '7504' '7507' '7509') then gastro=1;

if &icdvar{i} in:('7511' '7512' '7513' '7514' '7515' '75160' '7518' '7519') then do; gastro=1; end;
if &icdvar{i} in:('75161' '75162' '75169' '7517' ) then do; gastro=1; progressive=1; end;

if &icdvar{i} in: ('75261' '75262' '7527') then genito=1;

if &icdvar{i} in:('7530' '7531' '7532') then do; genito=1; progressive=1; end;
if &icdvar{i} in:('7534' '7535' '7536' '7537' '7538' '7539') then genito=1;

if &icdvar{i} in:('7540' '7542'
                '75430' '75431' '75432' '75433' '75434' '75435'
                '7547'
                '7554' '75553' '75554' '75558') then musculo=1;

if &icdvar{i} in:('7560' '7561' '7562' '7563' '7564' '7565' '75683') then musculo=1;
if &icdvar{i} in:('7566' '7567') then do; musculo=1; progressive=1; end;

if &icdvar{i} in:('7570' '7571' '75731') then derm=1;

if &icdvar{i} in:('7580' '7583' '7585' '7586' '7587' '7588' '7589') then genetic=1;
if &icdvar{i} in:('7581' '7582') then do; genetic=1; progressive=1; end;

if &icdvar{i} =:'759' then
    do;
        genetic=1;
        if &icdvar{i} in:('7590' '7591' '7592' '7593' '7594' '7596') then progressive=1;
    end;

if &icdvar{i} =:'7707' then pulresp=1;

if &icdvar{i} in:('78001' '78003') then do; neuro=1; progressive=1; end;
if &icdvar{i} in:('78051' '78053' '78057') then neuro=1;

if &icdvar{i} in:('887' '896' '897') then musculo=1;

```

```

if &icdvar{i} =:'952'      then do;  neuro    =1; progressive=1; end;

if &icdvar{i} =:'V08'     then do;  immuno   =1; progressive=1; end;
if &icdvar{i} =:'V420'    then do;  renal    =1; progressive=1; end;
if &icdvar{i} =:'V421'    then do;  cardiac  =1; progressive=1; end;
if &icdvar{i} =:'V422'    then do;  cardiac  =1;                end;
if &icdvar{i} =:'V426'    then do;  pulresp  =1; progressive=1; end;

if &icdvar{i} in:('V427' 'V4284') then do;  gastro=1; progressive=1; end;

if &icdvar{i} =:'V4281'   then do;  hemato   =1; progressive=1; end;
if &icdvar{i} =:'V4283'   then do;  endo     =1; progressive=1; end;
if &icdvar{i} =:'V4322'   then do;  cardiac  =1; progressive=1; end;

if &icdvar{i} in:('V520' 'V521') then musculo=1;

end;
if last.&claimid then output;
run;

/* Roll up to one record per child, with a single flag for each body type, a sum across claims for
each body type, and presence of a progressive condition or malignancy.
Calculate final condition determinations. */

data oput.results_pmca_v1_0(keep=&sid
                        cond_less
                        cond_more);

set flagclaims;
by &sid;
retain  anycardiac anycranio anyderm anyendo anygastro anygenetic anygenito anyhemato anyimmuno anymalign
        anymetab  anymusculo anyneuro anyopthal anyotol  anyotolar anypulresp anyrenal anymh
        anyprogressive

        anycardiac2 anycranio2 anyderm2 anyendo2 anygastro2 anygenetic2 anygenito2 anyhemato2 anyimmuno2 anymalign2
        anymetab2  anymusculo2 anyneuro2 anyopthal2 anyotol2  anyotolar2 anypulresp2 anyrenal2 anymh2

        cardiac2h  cranio2h  derm2h  endo2h  gastro2h  genetic2h  genito2h  hemato2h  immuno2h  malign2h
        metab2h  musculo2h  neuro2h  opthal2h  otol2h  otolar2h  pulresp2h  renal2h  mh2h  ;

if first.&sid then
do;
        anycardiac= 0; anycranio      = 0; anyderm          = 0; anyendo      = 0; anygastro  = 0; anygenetic = 0;

```



```
anygenito = 0; anyhemato = 0; anyimmuno = 0; anyalign = 0; anymetab = 0; anymusculo = 0;
anyneuro = 0; anyopthal = 0; anyotol = 0; anyotolar = 0; anypulresp = 0; anyrenal = 0;
anymh = 0; anyprogressive = 0;
```

```
anycardiac2 = 0; anycranio2 = 0; anyderm2 = 0; anyendo2 = 0; anygastro2 = 0; anygenetic2 = 0;
anygenito2 = 0; anyhemato2 = 0; anyimmuno2 = 0; anymalign2 = 0; anymetab2 = 0; anymusculo2 = 0;
anyneuro2 = 0; anyopthal2 = 0; anyotol2 = 0; anyotolar2 = 0; anypulresp2 = 0; anyrenal2 = 0;
anymh2 = 0;
```

```
cardiac2h = 0; cranio2h = 0; derm2h = 0; endo2h = 0; gastro2h = 0; genetic2h = 0;
genito2h = 0; hemato2h = 0; immuno2h = 0; malign2h = 0; metab2h = 0; musculo2h = 0;
neuro2h = 0; opthal2h = 0; otol2h = 0; otolar2h = 0; pulresp2h = 0; renal2h = 0;
mh2h = 0;
```

end;

*if a body system is indicated, create

- 1) a flag indicating the presence of that body system involvement (indicator y/n), and
- 2) the number of claims with that body system indicated (sum across claims);

```
                *indicator y/n;                *sum across claims;
if cardiac      = 1 then do; anycardiac      = 1; anycardiac2 + 1; end;
if cranio       = 1 then do; anycranio       = 1; anycranio2  + 1; end;
if derm         = 1 then do; anyderm         = 1; anyderm2   + 1; end;
if endo         = 1 then do; anyendo         = 1; anyendo2   + 1; end;
if gastro       = 1 then do; anygastro       = 1; anygastro2 + 1; end;
if genetic      = 1 then do; anygenetic      = 1; anygenetic2 + 1; end;
if genito       = 1 then do; anygenito       = 1; anygenito2  + 1; end;
if hemato       = 1 then do; anyhemato       = 1; anyhemato2  + 1; end;
if immuno       = 1 then do; anyimmuno       = 1; anyimmuno2  + 1; end;
if metab        = 1 then do; anymetab        = 1; anymetab2   + 1; end;
if musculo      = 1 then do; anymusculo      = 1; anymusculo2 + 1; end;
if neuro        = 1 then do; anyneuro        = 1; anyneuro2   + 1; end;
if pulresp      = 1 then do; anypulresp      = 1; anypulresp2 + 1; end;
if renal        = 1 then do; anyrenal        = 1; anyrenal2   + 1; end;
if opthal       = 1 then do; anyopthal       = 1; anyopthal2  + 1; end;
if otol         = 1 then do; anyotol         = 1; anyotol2   + 1; end;
if otolar       = 1 then do; anyotolar       = 1; anyotolar2  + 1; end;
if mh           = 1 then do; anymh           = 1; anymh2     + 1; end;
if progressive  = 1 then do; anyprogressive  = 1;                end;
if malign       = 1 then do; anymalign       = 1;                end;
```

*roll up to last observation;

```
if last.&sid then
do;
```

```
length cond_less cond_more $24.;
```

```
*****
CONDITION DETERMINATION--calculate condition type based on two different algorithms
*****;
```

```
*-----
LESS CONSERVATIVE ALGORITHM
```

```
-----
The less conservative version (cond_less) calculates values as
```

```
'Complex Chronic': 1) more than one body system is involved, or
                   2) one or more conditions are progressive, or
                   3) one or more conditions are malignant
```

```
'Non-complex Chronic': 1) only one body system is involved, and
                       2) the condition is not progressive or malignant
```

```
'Non-Chronic':      1) no body system indicators are present, and
                       2) the condition is not progressive or malignant
```

```
*count number of different body systems involved;
```

```
scount_less = anycardiac + anycranio + anyderm  + anyendo  + anygastro + anygenetic +
              anygenito  + anyhemato + anyimmuno + anymetab + anymusculo + anyneuro  +
              anypulresp + anyrenal  + anyopthal + anyotol  + anyotolar + anymh;
```

```
*set condition based on less conservative algorithm;
```

```
if scount_less    >= 2 or
   anyprogressive = 1 or
   anymalign      = 1      then cond_less = '3 Complex Chronic';   else
if scount_less    = 1      then cond_less = '2 Non-complex Chronic'; else
                           cond_less = '1 Non-Chronic';
```

```
*-----
MORE CONSERVATIVE ALGORITHM
```

```
-----
The more conservative version (cond_more) calculates values as
```

'Complex Chronic': 1) more than one body system is involved,
and each must be indicated in more than one claim, or
2) one or more conditions are progressive, or
3) one or more conditions are malignant

'Non-complex Chronic': 1) only one body system is indicated in more than one claim, and
2) the condition is not progressive or malignant

'Non-Chronic': 1) no body system indicators are present in more than one claim, and
2) the condition is not progressive or malignant

*identify body systems with indications in at least two different claims;

```
if anycardiac2 >= 2 then cardiac2h = 1;
if anycranio2 >= 2 then cranio2h = 1;
if anyderm2 >= 2 then derm2h = 1;
if anyendo2 >= 2 then endo2h = 1;
if anygastro2 >= 2 then gastro2h = 1;
if anygenetic2 >= 2 then genetic2h = 1;
if anygenito2 >= 2 then genito2h = 1;
if anyhemato2 >= 2 then hemato2h = 1;
if anyimmuno2 >= 2 then immuno2h = 1;
if anymetab2 >= 2 then metab2h = 1;
if anymusculo2 >= 2 then musculo2h = 1;
if anyneuro2 >= 2 then neuro2h = 1;
if anyopthal2 >= 2 then opthal2h = 1;
if anyotol2 >= 2 then otol2h = 1;
if anyotolar2 >= 2 then otolar2h = 1;
if anypulresp2 >= 2 then pulresp2h = 1;
if anyrenal2 >= 2 then renal2h = 1;
if anymh2 >= 2 then mh2h = 1;
```

* count number of body systems that are indicated in more than one claim ;

```
scount_more = cardiac2h + cranio2h + derm2h + endo2h + gastro2h + genetic2h + genito2h +
             hemato2h + immuno2h + metab2h + musculo2h + neuro2h + pulresp2h + renal2h +
             otol2h + otolar2h + opthal2h + mh2h;
```

*set condition based on more conservative algorithm;

```
if scount_more >= 2 or
   anyprogressive = 1 or
   anymalign = 1 then cond_more = '3 Complex Chronic'; else
if scount_more = 1 then cond_more = '2 Non-complex Chronic'; else
```

```
cond_more = '1 Non-Chronic';
```

```
    output;
```

```
end;
```

```
run;
```