```
******************************************************************************

    UTHSC

    PROGRAM NAME        : MUI 3.1 star rating- high risk medication in the
elderly

    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :The patient’s measurement period begins on the date
of the first fill of the target
                         medication (i.e., index date) and extends through
the last day of the enrollment
                         period or until death or disenrollment. The index
date should occur at least 91 days
                         before the end of the enrollment period.
                         Patients who received at least two prescription
fills for the same high-risk medication
                         (Table HRM-A: High-Risk Medications) during the
measurement period.

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

******************************************************************************
***

    REQUIREMENTS:

******************************************************************************
***

    Ver#  Author & Program History Description      Peer reviewer

******************************************************************************
***

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

******************************************************************************
***;

/*The percentage of patients 65 years of age and older who received two or
more prescription fills for a high-risk medication during the measurement
period.—2013 PQA technical note*/

DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';

/*******************************************************
*******************************************************
```

```
**************hrm for 2013**************************
***************2 fills of medication**************************************
*******************************************************/

proc import datafile="F:\jwang26\Part D data\hrm1.csv"
/*This drug list is obtained from PQA NDC list of 2013 */
     out=hrm
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;


proc contents data=hrm short; run;
/*Amitriptyline_flag Amobarbital_flag Benztropine_flag Brand_Drug_Name
Brompheniramine_flag Butabarbital_flag Butalbital_flag
Carbinoxamine_flag Carisoprodol_flag Chloral_hydrate_flag
Chlorpheniramine_flag Chlorpropamide_flag Chlorzoxazone_flag
Clemastine_flag Clomipramine_flag Cyclobenzaprine_flag Cyproheptadine_flag
Dexchlorpheniramine_flag Digoxin_flag
Diphenhydramine_flag Dipyridamole_flag Disopyramide_flag Doxepin_flag
Doxylamine_flag DrugStrength Ergoloid_mesylates_flag
Estrogens_flag Eszopiclone_flag Generic_Drug_Name Glyburide_flag
Guanfacine_flag Hydroxyzine_flag Imipramine_flag Indomethacin_flag
Isoxsuprine_flag Ketorolac_flag Megestrol_flag Meperidine_flag
Meprobamate_flag Metaxalone_flag Methocarbamol_flag Methyldopa_flag
NDC New_Update_Flag Nifedipine_flag Nitrofurantoin_flag Obsolete_Date
Orphenadrine_flag Pentazocine_flag Pentobarbital_flag
Phenobarbital_flag Promethazine_flag Reserpine_flag Route Secobarbital_flag
Therapeutic_Category Thioridazine_flag Thyroid_flag
Ticlopidine_flag Trihexyphenidyl_flag Trimethobenzamide_flag
Trimipramine_flag Triprolidine_flag Zaleplon_flag Zolpidem_flag
*******************************************************;*/

data hrm1;
set hrm;
if Amitriptyline_flag='X' then lable=1;
if Amobarbital_flag ='X' then lable=2;
if Benztropine_flag='X' then lable=3;
if Brompheniramine_flag ='X' then lable=4;
if Butabarbital_flag ='X' then lable=5;
if Butalbital_flag ='X' then lable=6;
if Carbinoxamine_flag ='X' then lable=7;
if Carisoprodol_flag ='X' then lable=8;
if Chloral_hydrate_flag ='X' then lable=9;
if Chlorpheniramine_flag ='X' then lable=10;
if Chlorpropamide_flag ='X' then lable=11;
if Chlorzoxazone_flag ='X' then lable=12;
if Clemastine_flag ='X' then lable=13;
if Clomipramine_flag ='X' then lable=14;
if Cyclobenzaprine_flag ='X' then lable=15;
if Cyproheptadine_flag ='X' then lable=16;
if Dexchlorpheniramine_flag ='X' then lable=17;
if Digoxin_flag ='X' then lable=18;
if Diphenhydramine_flag ='X' then lable=19;
if Dipyridamole_flag ='X' then lable=20;
if Disopyramide_flag ='X' then lable=21;
if Doxepin_flag ='X' then lable=22;
if Doxylamine_flag ='X' then lable=23;
if DrugStrength ='X' then lable=24;
if Ergoloid_mesylates_flag ='X' then lable=25;
if Estrogens_flag ='X' then lable=26;
if Eszopiclone_flag ='X' then lable=27;
if Glyburide_flag ='X' then lable=28;
if Guanfacine_flag ='X' then lable=29;
if Hydroxyzine_flag ='X' then lable=30;
if Imipramine_flag ='X' then lable=31;
if Indomethacin_flag ='X' then lable=32;
if Isoxsuprine_flag ='X' then lable=33;
if Ketorolac_flag ='X' then lable=34;
if Megestrol_flag ='X' then lable=35;
if Meperidine_flag ='X' then lable=36;
if Meprobamate_flag ='X' then lable=37;
if Metaxalone_flag ='X' then lable=38;
if Methocarbamol_flag ='X' then lable=39;
if Methyldopa_flag='X' then lable=40;
if Nifedipine_flag='X' then lable=41;
if Nitrofurantoin_flag ='X' then lable=42;
if Orphenadrine_flag ='X' then lable=43;
if Pentazocine_flag ='X' then lable=44;
if Pentobarbital_flag='X' then lable=45;
if Phenobarbital_flag ='X' then lable=46;
if Promethazine_flag ='X' then lable=47;
if Reserpine_flag='X' then lable=48;
if Secobarbital_flag ='X' then lable=49;
if Thioridazine_flag ='X' then lable=50;
if Thyroid_flag ='X' then lable=51;
if Ticlopidine_flag='X' then lable=52;
if Trihexyphenidyl_flag ='X' then lable=53;
if Trimethobenzamide_flag ='X' then lable=54;
if Trimipramine_flag ='X' then lable=55;
if Triprolidine_flag ='X' then lable=56;
if Zaleplon_flag ='X' then lable=57;
if Zolpidem_flag='X' then lable=58;
if Therapeutic_Category='*Non-Benzodiazepine - GABA-Receptor Modulators***'
then nonben=1; else nonben=0;
keep ndc lable nonben;
run;

proc sort data=hrm1; by ndc;
run;


data pde2013hrm;
set ccw2.pde2013 ;/*this is 2013 Medicare Part D Event data*/
ndc=PROD_SRVC_ID*1;
keep bene_id ndc str SRVC_DT QTY_DSPNSD_NUM DAYS_SUPLY_NUM;
run;

data ccw2.pde2013hrm;
set pde2013hrm;
run;
```

```
proc sort data=pde2013hrm out=a; by ndc;
run;

data b;
merge hrm1(in=in1) a(in=in2);
by ndc;
if in1 and in2;
run;

proc sort data=b out=c; by bene_id lable SRVC_DT;
run;

proc freq data=c noprint;
tables bene_id*lable/out=d;
run;


data e;
set d;
keep bene_id lable;
if count>=2;
run;

proc sort data=e out=f nodupkey; by bene_id  ;
run;

proc sort data=e out=f1; by bene_id lable ;
run;

data h;
merge f1(in=in1) c(in=in2);
by bene_id lable;
if in1 and in2;
run;


/*first  should be 90 days before 12/31/2013*/

proc sort data=h out=i nodupkey; by bene_id lable ;
run;

data j;
set i;
if '31Dec2013'd-SRVC_DT>90;
run;


data j1;
set j;
by bene_id;
if first.bene_id then output;
run;

data ccw2.hrm1;
set j1;
if lable=42 or lable= 48 or lable=18 or lable=22 or lable=27 or lable = 57 or
lable = 58 then hrm=0;else hrm=1;
```

```
keep bene_id hrm;
run;


data j2;
merge j(in=in1) c(in=in2);
by bene_id lable;
if in1 and in2;
run;

data ccw2.j2;
set j2;
run;

/*A. For nitrofurantoin, a patient is included in the numerator if he/she
received at least two prescription fills for the
medication andif the cumulative days supply for any nitrofurantoin product is
greater than 90 days during the
measurement period.*/



data nitro1;
set j2;
if lable=42;
run;

proc means data=nitro1 noprint;
class bene_id;
var DAYS_SUPLY_NUM;
output out=nitro2 sum= ;
run;


data nitro3;
set nitro2;
if DAYS_SUPLY_NUM>90;
if bene_id ne '';
run;

data ccw2.hrm2;
set nitro3;
hrm=1;
run;

/* Methods are similar for the below measures*/
B. For reserpine, a patient is included in the numerator if he/she received
at least two prescription fills for the
medication andif the average daily dose is greater than 0.1mg.
*/

/*For digoxin, a patient is included in the numerator if he/she received at
least two prescription fills for the
medication and if the average dailydose is greater than 0.125mg. */

/*For doxepin, a patient is included in the numerator if he/she received at
least two prescription fills for the
```

```
medication andif the average daily dose is greater than 6mg.*/



/*The cumulative calculation applies to theclass of nonbenzodiazepine
hypnotics and not for each individual
medication. A patient is included in the numerator if he/she received at
least two prescription fills for any
medication in the class andif the cumulative days supply for any product is
greater than 90 days during the
measurement period. . For example, if a patient received a 30 day supply of
zolpidem, a second fill for 30 days
supply of zolpidem and then a fill for 35 days supply eszopiclone (all during
the measurement period), this would
qualify for inclusion in the numerator. */


data nonben1;
set j2;
if lable=27 or lable=57 or lable=58;
run;
proc sort data=nonben1 out=nonben2 nodupkey; by bene_id;
run;
data nonben3;
set c;
if lable=27 or lable=57 or lable=58;
run;

proc means data=nonben3 noprint;
class bene_id ;
var DAYS_SUPLY_NUM;
output out=nonben4 sum= ;
run;

data nonben5;
set nonben4;
if DAYS_SUPLY_NUM>90;
if bene_id ne '';
run;

proc sort data=nonben5 out=nonben6 ; by bene_id;
run;
data nonben7;
merge nonben2(in=in1) nonben6(in=in2);
by bene_id;
if in1 and in2;
keep bene_id;
run;

data ccw2.hrm6;
set nonben7;
hrm=1;
keep bene_id hrm;
run;


proc sort data=ccw2.hrm1; by bene_id;run;
proc sort data=ccw2.hrm2; by bene_id;run;
proc sort data=ccw2.hrm3; by bene_id;run;
proc sort data=ccw2.hrm4; by bene_id;run;
proc sort data=ccw2.hrm5; by bene_id;run;
proc sort data=ccw2.hrm6; by bene_id;run;

data ccw2.hrm7;
merge ccw2.hrm1 ccw2.hrm2 ccw2.hrm3 ccw2.hrm4 ccw2.hrm5 ccw2.hrm6;
by bene_id;
run;

proc sort data=ccw2.hrm7 out=ccw2.hrm8; by bene_id;run;
proc sort data=ccw2.mbsf1 out=mbsf1age nodupkey; by bene_id;run;

proc sort data=ccw2.pde2013 out=hrmdrugnum; by bene_id GNN; run;
proc freq data=hrmdrugnum noprint;
tables bene_id*GNN/out=hrmdrugnum;
run;

data hrmdrugnum1;
set hrmdrugnum;
if count>=2;
run;

proc sort data=hrmdrugnum1 out=hrmdrugnum2 nodupkey; by bene_id;run;
data ccw2.hrm9;
merge ccw2.hrm8(in=in1) mbsf1age(in=in2);
by bene_id;
if age>=65;
if hrm=. then hrm=0;
run;

data hrm9;
merge ccw2.hrm9(in=in1) hrmdrugnum2(in=in2);
by bene_id;
if in2;
if age>=66;
if hrm=. then hrm=0;
run;

data ccw2.hrm10;
set hrm9;
by bene_id;
keep bene_id hrm;
run;
```

```
*****************************************************************************
*******************
    UTHSC

    PROGRAM NAME        :MUI 3.2: star rating- appropriate treatment

    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

*****************************************************************************
***

    REQUIREMENTS:

*****************************************************************************
***

    Ver#  Author & Program History Description      Peer reviewer

*****************************************************************************
***

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

*****************************************************************************
***;


DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';


/**Patients who were dispensed at least one prescription for an oral
hypoglycemic agent, insulin, incretin mimetics (Table APP-A: Oral
Hypoglycemic, Insulin, and Incretin Mimetics) and at least one prescription
for an antihypertensive agent (Table APP-B: Antihypertensive Agents) during
the measurement year.********/

proc import datafile="F:\jwang26\Part D data\hyperdiabetesall.csv"
/*this drug list is from PQA NDC list for medications*/
     out=hyperdiabetesall
     dbms=csv
        replace;
     getnames=yes;
       guessingrows=32767;
run;
```

```
proc sort data=hyperdiabetesall; by ndc;
run;
proc sort data=ccw2.pde2013hrm; by ndc;run;
*******************************************************;
data b;
merge ccw2.pde2013hrm(in=in1)  hyperdiabetesall(in=in2);
by ndc;
if in1 and in2;
run;

proc sort data=b out=c; by bene_id ndc SRVC_DT label;
run;

proc freq data=c noprint;
tables bene_id*label/out=d;
run;

proc means data=d noprint;
class bene_id;
var label;
output out=e sum=;
run;
data f;
set e;
keep bene_id label;
if label=3;
run;




/*****************************************************
*******The number of patients who receive an ACEI/ARB/direct renin inhibitor
or ACEI/ARB/direct renin inhibitor combination during the measurement year.
Refer to Table APP-C: Renin Angiotensin System Antagonists**************/

proc import datafile="F:\jwang26\Part D data\rasa.csv"
/*this drug list is from PQA NDC list for medications*/
     out=hyperdiabetes
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=hyperdiabetes; by ndc;
run;
data a1;
set hyperdiabetes;
label2=0;
keep ndc label2;
run;


data b1;
merge ccw2.pde2013hrm(in=in1)  a1(in=in2);
by ndc;
if in1 and in2;
```

```
run;


proc sort data=b1 out=c1 nodupkey; by bene_id; run;

data ccw2.appropriate1;
merge f(in=in1) c1(in=in2);
by bene_id;
if in1 and in2;
appropriate=1;
keep bene_id appropriate;
run;

data ccw2.appropriate2;
merge  ccw2.appropriate1 f;
by bene_id;
if  appropriate=. then appropriate=0;
run;
```

```
******************************************************************************
    UTHSC

    PROGRAM NAME        :MUI 3.3: star rating- PDC-Proportion of Days Covered

    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

******************************************************************************

    REQUIREMENTS:

******************************************************************************

    Ver#  Author & Program History Description      Peer reviewer

******************************************************************************

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

*****************************************************************************;
/*Reference:Measuring Medication Adherence with Simple Drug Use and
Medication Switching. Stacy Wang, Walgreens, Deerfield, IL etc.*/
/*This piece of code was written for PDC calculation or verification. */

/*includes:
PDC <80% for any of the three drug classes: renin-angiotensin system
antagonists, cholesterol medications among adults with coronary artery
disease, and oral diabetes medications;
PDC <90% for HIV antiretroviral medications;
PDC <80% for beta-blockers, calcium-channel blockers, and non-warfarin oral
anticoagulants;

/*****************************************************************************
** Macro Parameters: ***
** input - Input Dataset name ***
** output - Output Dataset name ***
** pid - Unique patient identifier ***
** fill_dt - Prescription fill date ***
** dos - Fill days of supply ***
** start_dt - Start date of study period ***
** end_dt - End date of study period ***
*****************************************************************************/


/*****************************************************
******************************************************
***************pdc FOR RASA***************************
```

```
******************************************************
********************************************************/
/*PQA technical note 2013: ACEI/ARB/Direct Renin Inhibitor or ACEI/ARB/Direct
Renin Inhibitor Combination (Table PDC-B: RAS Antagonists) on two unique
dates of service during the measurement period*/

DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';
proc import datafile="F:\jwang26\Part D data\rasa.csv"
/* this data set is from PQA NDC list for medications of 2013*/
     out=rasa
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=rasa; by ndc;
run;

********************************************************;

data a;
set ccw2.pde2013;/*Medicare Part D Event data 2013*/
ndc=PROD_SRVC_ID*1;
drop PROD_SRVC_ID;
run;

proc sort data=a; by ndc ;
run;

data ccw2.pdcdrug;
set a;
run;

data b;
merge ccw2.pdcdrug(in=in1) rasa(in=in2);
by ndc;
if in1 and in2;
run;

proc sort data=b out=b; by bene_id ;run;

proc sort data=b out=b1 nodupkey; by bene_id SRVC_DT ;
run;
proc freq data=b1 noprint;
table bene_id/out=b2; run;

data b3;
set b2;
if count>=2;
keep bene_id;
run;
**************90 days before the measure period*********;
data b4;
set b;
```

```
by bene_id;
if first.bene_id then output;
run;

data b5;
set b4;
if '31DEC2013'd-srvc_dt>90;
run;


data pdcrasa;
merge  b(in=in1) b3(in=in2) b5(in=in3);
by bene_id;
if in2 and in3;
run;

proc sort data=pdcrasa out=ccw2.pdcrasa;
by bene_id SRVC_DT;run;



data claims;
set ccw2.pdcrasa;
patient_id=bene_id;
fill_date=SRVC_DT;
format fill_date DDMMYY10. ;
days_supply=DAYS_SUPLY_NUM;
keep patient_id  fill_date  days_supply;
run;

%macro PDC_Single(
input=,
output=,
pid=,
fill_dt=,
dos=,
start_dt=,
end_dt=);

** Step 1: Exclude fills outside of study period **;
data &input.;
set &input.;
if &end_dt. >= &fill_dt >= &start_dt.;
run;

** Step 2: Find out the maximum number of fills a patient can have **;
** and save it into macro variable, &_mcount **;
proc sql noprint;
     create table temp as
     select &pid., count(*) as ct
     from &input.
     group by &pid.;
     select max(ct) into :_mcount from temp;
quit;
%let _mcount=&_mcount;
```

```
** Step 3: Transpose dataset **;
proc transpose data=&input. out=fill_dates(drop=_name_) prefix=fill_dt;
      by &pid.;
      var &fill_dt.;
run;
proc transpose data=&input. out=fill_DOS(drop=_name_) prefix=days_sply;
      by &pid.;
      var &dos.;
run;

data claims_refmt;
      merge fill_dates fill_DOS;
      by &pid.;
      start_study_dt=fill_dt1;
      end_study_dt=&end_dt.;
      format start_study_dt end_study_dt mmddyy10.;
      temp=&end_dt. - fill_dt1+1;
run;



** Step 4: Adjust prescription fill date for early refills **;
data pdc1(keep=&pid. adj_fill_date days_supply end_study_dt);
set claims_refmt;
      array dates(*) fill_dt1 - fill_dt&_mcount.;
      array dos(*) days_sply1 - days_sply&_mcount.;

      adj_fill_date=dates(1);
      days_supply=dos(1);
      output;

      do i = 2 to dim(dates) while (dates(i) ne .);/*array dates(*) fill_dt1
- fill_dt&_mcount.;*/
      if dates(i) <dates(i-1)+dos(i-1)/*date 2<date1+dose1 not finish and new
prescription*/
      then do;
            dates(i) =dates(i-1)+dos(i-1);
        adj_fill_date=dates(i);
        days_supply=dos(i);
            end;
        adj_fill_date=dates(i);/*dates(i) >dates(i-1)+dos(i-1)*/
        days_supply=dos(i);
            output;
        end;
        format adj_fill_date mmddyy10.;
run;


** Step 5: Truncate fills fall outside of study period **;
data pdc2;
      set pdc1;
      if adj_fill_date <=end_study_dt then do;/*end_study_dt=&end_dt.;*/
        if adj_fill_date + days_supply >end_study_dt then do;
            days_supply=end_study_dt-adj_fill_date+1;
        end;
        output;
      end;
```

```
run;

** Step 6: Calculate PDC at patient level **;
proc sql;
create table coverdays as
select &pid.,
sum(days_supply) as dayscovered
from pdc2
group by &pid.;
quit;

data &output.;
merge claims_refmt coverdays;
by patient_id;
pdc=dayscovered/temp;
keep patient_id start_study_dt end_study_dt temp dayscovered pdc;
run;



%mend PDC_Single;
** sample macro call **;
%PDC_single(
input=claims,
output=pdc,
pid=patient_id,
fill_dt=fill_date,
dos=days_supply,
start_dt='01JAN2013'd,
end_dt='31DEC2013'd);

data ccw2.rasapdc;
set pdc;
if pdc<0.8 then rasapdc=1;
else rasapdc=0;
bene_id=patient_id;
keep bene_id rasapdc;
run;




/*One example provided. And the programs for other PDC measure are the
same.*/
```

```
******************************************************************************
    UTHSC

    PROGRAM NAME        :MUI 3.4: star rating- Drug Drug INTERACTION
    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

******************************************************************************

    REQUIREMENTS:

******************************************************************************

    Ver#  Author & Program History Description      Peer reviewer

******************************************************************************

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

*****************************************************************************
PQA: Patient received a target medication also was dispensed another
concurren precipitant medication during the measurement period. (Table DDI-A:
Target Medications and Precipitant Medications. ;

DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';



proc import datafile="F:\jwang26\Part D data\did.csv"****PQA NDC
list***********;
     out=did
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=did; by ndc;
run;

***************************************;
data did2;
merge ccw2.pdcdrug(in=in1) did (in=in2);
by ndc;
if in1 and in2;
run;
```

```
proc sort data=did2 out=ccw2.did3; by bene_id category step  SRVC_DT;
run;

/*
proc sort data=didy1 out=test nodupkey;by bene_id category; run;56*/


data didy1;
set ccw2.did3;
if step=1;
keep bene_id category step DAYS_SUPLY_NUM QTY_DSPNSD_NUM SRVC_DT;
run;

proc sort data=didy1; by category bene_id SRVC_DT DAYS_SUPLY_NUM;run;
/*
proc sort data=test; by  bene_id SRVC_DT;run;*/

data  didy2;
        format end_dt1 end_dt2 max_dt pmax_dt mmddyy8.;
        set didy1;
          by category bene_id SRVC_DT;
               /* reset stats for first id record */
            retain pmax_dt totdays  indexdate/*indexdate for each course*/;

            if first.bene_id then do;
                        totdays = 0;
                        end_dt1 = .;
                end_dt2 = .;
                        pmax_dt = .;
                      indexdate = SRVC_DT;
                  end;

                  /* determine the last day of Rx based on later of: */
                  /* Rx date + days supply or total days from index date */
                  /* based on all Rx */

                  /* if greater than 1 day gap then consider to have stopped
*/

                  if SRVC_DT - pmax_dt >1 then do;/*gap=1*/
                     indexdate = SRVC_DT;
                     totdays = 0;


                  end;

                  end_dt1 = SRVC_DT+DAYS_SUPLY_NUM;
                  totdays + DAYS_SUPLY_NUM;
                  end_dt2 =indexdate + totdays;
                  max_dt = max(end_dt1, end_dt2);
                  pmax_dt = max_dt;

                  format indexdate mmddyy8.;

run;
```

```
data didy3;
set didy2;
keep category bene_id SRVC_DT pmax_dt;
run;


/************************************************************************
************************step 2 drug***********************
***********************************************/
data didz1;
set ccw2.did3;
if step=2;
keep bene_id category step DAYS_SUPLY_NUM QTY_DSPNSD_NUM SRVC_DT;
run;

proc sort data=didz1; by category bene_id SRVC_DT DAYS_SUPLY_NUM;run;

data  didz2;
        format end_dt1 end_dt2 max_dt pmax_dt mmddyy8.;
        set didz1;
          by category bene_id SRVC_DT;
               /* reset stats for first id record */
            retain pmax_dt totdays  indexdate/*indexdate for each course*/;

            if first.bene_id then do;
                        totdays = 0;
                        end_dt1 = .;
                end_dt2 = .;
                        pmax_dt = .;
                      indexdate = SRVC_DT;
                  end;

                  /* determine the last day of Rx based on later of: */
                  /* Rx date + days supply or total days from index date */
                  /* based on all Rx */

                  /* if greater than 1 day gap then consider to have stopped
*/

                  if SRVC_DT - pmax_dt >1 then do;/*gap=1*/
                     indexdate = SRVC_DT;
                     totdays = 0;


                  end;

                  end_dt1 = SRVC_DT+DAYS_SUPLY_NUM;
                  totdays + DAYS_SUPLY_NUM;
                  end_dt2 =indexdate + totdays;
                  max_dt = max(end_dt1, end_dt2);
                  pmax_dt = max_dt;

                  format indexdate mmddyy8.;

run;

/*******Find the overlap days*************/
```

```
******************************************************************************
    UTHSC

    PROGRAM NAME        :MUI 3.5: star rating- excessive dose
    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

******************************************************************************

    REQUIREMENTS:

******************************************************************************

    Ver#  Author & Program History Description      Peer reviewer

******************************************************************************

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

*****************************************************************************;
PQA: This measure was retired—2016;

dM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';


proc import datafile="F:\jwang26\Part D data\excessivedose.csv"/*PQA
technical notes and NDC list*/
     out=excessivedose
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;


proc contents data=excessivedose short; run;
/*
Alphabetic List of Variables for WORK.EXCESSIVEDOSE
BIGUANIDE_FLAG BRAND_DRUG_NAME DDPIV_FLAG DRUGSTRENGTH1 DRUGSTRENGTH2
GENERIC_DRUG_NAME MAXDOSE1 MAXDOSE2 NDC NEW_UPDATE_FLAG
OBSOLETE_DATE RECYCLED_NDC_FLAG ROUTE SFU_FLAG THERAPEUTIC_CATEGORY TZD_FLAG
*/

data sfu;
set excessivedose;
if sfu_flag ne "";
if SFU_FLAG="Drug1" then strength=DRUGSTRENGTH1 ;
if SFU_FLAG="Drug2" then strength=DRUGSTRENGTH2 ;

if SFU_FLAG="Drug1" then  maxdose=maxdose1;
if SFU_FLAG="Drug2" then  maxdose=maxdose2;
keep ndc  sfu_flag strength maxdose;
run;

data BIGUANIDE;
set excessivedose;
if BIGUANIDE_FLAG ne "";
if BIGUANIDE_FLAG="Drug1" then strength=DRUGSTRENGTH1 ;
if BIGUANIDE_FLAG="Drug2" then strength=DRUGSTRENGTH2 ;

if BIGUANIDE_FLAG="Drug1" then  maxdose=maxdose1;
if BIGUANIDE_FLAG="Drug2" then  maxdose=maxdose2;
keep ndc  BIGUANIDE_FLAG strength maxdose;
run;

data tzd;
set excessivedose;
if TZD_FLAG ne "";
if TZD_FLAG="Drug1" then strength=DRUGSTRENGTH1 ;
if TZD_FLAG="Drug2" then strength=DRUGSTRENGTH2 ;

if TZD_FLAG="Drug1" then  maxdose=maxdose1;
if TZD_FLAG="Drug2" then  maxdose=maxdose2;
keep ndc  TZD_FLAG strength maxdose;
run;

data DDPIV;
set excessivedose;
if DDPIV_FLAG ne "";
if DDPIV_FLAG="Drug1" then strength=DRUGSTRENGTH1 ;
if DDPIV_FLAG="Drug2" then strength=DRUGSTRENGTH2 ;

if DDPIV_FLAG="Drug1" then  maxdose=maxdose1;
if DDPIV_FLAG="Drug2" then  maxdose=maxdose2;
keep ndc  DDPIV_FLAG strength maxdose;
run;





/*****************************************************
******************************************************
***************over dose for sfu***************************
*****************************************************
*******************************************************/


proc sort data=sfu; by ndc;run;

data a;
merge ccw2.pdcdrug(in=in1) sfu(in=in2);
by ndc;
```

```
if in1 and in2;
run;

proc sort data=a; by bene_id SRVC_DT;
run;


/*first  should be 90 days before 12/31/2013*/

data a1;
set a;
by bene_id;
if first.bene_id then output;
run;

data a2;
set a1;
if '31Dec2013'd-SRVC_DT>90;
keep bene_id;
run;

data a3;
merge a2(in=in1) a(in=in2);
by bene_id;
if in1;
run;

data excessivedosea1;
set a3;
dailydose=strength*QTY_DSPNSD_NUM/DAYS_SUPLY_NUM;
if dailydose>maxdose then snfoverflag=1;
else snfoverflag=0;
run;

proc sort data=excessivedosea1 out=excessivedosea2 nodupkey; by bene_id
snfoverflag;
run;

data ccw2.excessivedosea2;
set excessivedosea2;
keep bene_id  snfoverflag;
/*if snfoverflag=1;*/
run;



/*****************************************************
******************************************************
***************Methods for the below measure would be the same:
over dose for BIGUANIDE
over dose for tzd
over dose for ddpiv */
```

```
*****************************************************************************
    UTHSC

    PROGRAM NAME        :MUI 3.6: CMS- chronic use of atypical antipsychotics
by elderly beneficiaries in a nursing home
    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

*****************************************************************************

    REQUIREMENTS:

*****************************************************************************

    Ver#  Author & Program History Description      Peer reviewer

*****************************************************************************

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

****************************************************************************;



DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';

/*This part of code was obtained from independent variables: the step for
presparing variables for risk adjustment score
data ccw2.person;
set ccw2.mbsfsummary2013;
hicno=bene_id;
sex=BENE_SEX_IDENT_CD*1;
dob=BENE_BIRTH_DT;
orec=BENE_ENTLMT_RSN_ORIG*1;
keep bene_id hicno sex dob orec;
format dob mmddyy10.;
run;-one character, original reason for entitlement with the following
values:
            0 - OLD AGE (OASI)
                  1 - DISABILITY (DIB)
                  2 - ESRD
                  3 - BOTH DIB AND ESRD*/

data ageing;
set ccw2.person;
if orec=0;
```

```
bene_id=hicno;
keep bene_id;
run;

data snf1;
set ccw2.snfbase2013;
if NCH_BENE_DSCHRG_DT=. then NCH_BENE_DSCHRG_DT='31DEC2013'D;
keep bene_id CLM_ADMSN_DT NCH_BENE_DSCHRG_DT ;
run;



/****************************************************
******************************************************
*************** Atypical***************************
*****************************************************
*******************************************************/

/*6 month wash out period*/
proc import datafile="F:\jwang26\Part D data\atypical.csv"
     out=atypical
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=atypical; by ndc;
run;
*******************************************************;
data snf2;
merge ccw2.pdcdrug(in=in1) atypical(in=in2);
by ndc;
if in1 and in2;
keep bene_id srvc_dt DAYS_SUPLY_NUM;
run;

proc sort data=snf2; by bene_id SRVC_DT;
run;

data  snf3;
        format end_dt1 end_dt2 max_dt pmax_dt mmddyy8.;
        set snf2;
          by bene_id SRVC_DT;
               /* reset stats for first id record */
            retain pmax_dt totdays  indexdate/*indexdate for each course*/;

            if first.bene_id then do;
                        totdays = 0;
                        end_dt1 = .;
                end_dt2 = .;
                        pmax_dt = .;
                      indexdate = SRVC_DT;
                  end;

                  /* determine the last day of Rx based on later of: */
                  /* Rx date + days supply or total days from index date */
                  /* based on all Rx */
                  /* if greater than 1 day gap then consider to have stopped
*/

                  if SRVC_DT - pmax_dt >1 then do;/*gap=1*/
                     indexdate = SRVC_DT;
                     totdays = 0;

                  end;

                  end_dt1 = SRVC_DT+DAYS_SUPLY_NUM;
                  totdays + DAYS_SUPLY_NUM;
                  end_dt2 =indexdate + totdays;
                  max_dt = max(end_dt1, end_dt2);
                  pmax_dt = max_dt;

                  format indexdate mmddyy8.;

run;

data snf4;
set snf3;
drugend=pmax_dt;
format  drugend mmddyy8.;
keep bene_id SRVC_DT drugend ;
run;

proc sort data=snf4 out=snf4; by bene_id;run;
proc sort data=snf1 out=snf5 nodupkey; by bene_id CLM_ADMSN_DT
NCH_BENE_DSCHRG_DT ;run;


proc sql;
create table snfa
as select a.*,   b.drugend, b.SRVC_DT
from snf5 a, snf4 b
where   a.bene_id=b.bene_id ;
quit;

data snfb;
set snfa;
by bene_id;
if CLM_ADMSN_DT<=SRVC_DT and drugend<=NCH_BENE_DSCHRG_DT then days=drugend-
srvc_dt;/*startdate=SRVC_DT   finishdate=drugend */
if CLM_ADMSN_DT<=SRVC_DT and drugend>=NCH_BENE_DSCHRG_DT then
days=NCH_BENE_DSCHRG_DT-SRVC_DT;/*startdate=SRVC_DT
finishdate=NCH_BENE_DSCHRG_DT;*/
if SRVC_DT<=CLM_ADMSN_DT and drugend<=NCH_BENE_DSCHRG_DT then  days=drugend-
CLM_ADMSN_DT;/*startdate=CLM_ADMSN_DT; finishdate=drugend*/
if SRVC_DT<=CLM_ADMSN_DT and drugend>=NCH_BENE_DSCHRG_DT then
days=NCH_BENE_DSCHRG_DT-CLM_ADMSN_DT;/* startdate=CLM_ADMSN_DT
finishdate=NCH_BENE_DSCHRG_DT*/

run;

proc means data=snfb noprint;
class bene_id ;
```

```
var days;
output out=snfc sum= ;
run;

data atypical;
set snfc;
atypical=1;
if days>=90;
if bene_id ne '';
keep bene_id atypical;
run;

data atypical2;
merge atypical(in=in1) ageing(in=in2);
by bene_id;
if in1 and in2;
run;

data ccw2.atypical;
merge snf5(in=in1) ageing(in=in2) atypical2;
by bene_id;
if in1 and in2;
if atypical=. then atypical=0;
run;
```

```
*****************************************************************************
    UTHSC

    PROGRAM NAME        :MUI 3.7: PQA: Antipsychotic Use in Persons with
Dementia
    PROJECT NAME        :Finding Equitable and Effective MTM Eligibility
Criteria

    DESCRIPTION         :

    SOFTWARE VERSION    :SAS Windows 9.4

    RELATED PROGRAMS    :N/A

*****************************************************************************

    REQUIREMENTS:

*****************************************************************************

    Ver#  Author & Program History Description      Peer reviewer

*****************************************************************************

    001  Yanru Qiao                 production version of the program

    002  Dr. Junling Wang           Peer reviewer (Reviewed at Oct 3rd, 2016)

****************************************************************************
PQA: Event/Diagnosis (Denominator)
Identify all eligible patients with either:
1) A diagnosis of dementia (Table Dementia A)   and/ or
2) Individuals with two or more prescription claims within the measurement
year where the sum of days' supply is >60 for a cholinesterase
inhibitor or an NMDA receptor antagonist (Table Dementia B);


DM LOG 'CLEAR'; DM OUTPUT 'CLEAR';
* Output SAS Library *;
libname ccw 'F:\jwang26\CCW';
libname ccw2 'F:\jwang26\CCW2';

data dementiadiag;
set ccw2.d29;
bene_id=hicno;
if diag='2900' or diag='2901' or diag='29010' or diag='29011' or diag='29012'
or diag='29013' or diag='2902' or diag='29021' or diag='2903'
or diag='2904' or diag='29041' or diag='29042' or diag='29043' or
diag='29410' or diag='29420' or diag='3310' or diag='33182' ;
keep bene_id diag;
run;

proc sort data=dementiadiag out=ccw2.dementiadiag nodupkey;
by bene_id; run;


***********************************************************;
```

```
proc import datafile="F:\jwang26\Part D data\dementia.csv"/*FROM PQA NDC
list*/
     out=dementia
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=dementia; by ndc;
run;

data d;
merge ccw2.pdcdrug(in=in1) dementia(in=in2);
by ndc;
if in1 and in2;
run;

data d1;
set d;
if lable=1;
run;

data dd1;
set d;
if lable=2;
run;


proc sort data=d1 out=d2 ; by bene_id  SRVC_DT;run;
proc sort data=dd1 out=dd2 ; by bene_id  SRVC_DT;run;

proc freq data=d2 noprint;
table bene_id/out=d3; run;

data d4;
set d3;
if count>=2;
keep bene_id;
run;

proc means data=d2 noprint;
class bene_id ;
var DAYS_SUPLY_NUM;
output out=d5 sum= ;
run;

data d6;
set d5;
if DAYS_SUPLY_NUM>60;
if bene_id ne '';
run;

data d7;
merge d4(in=in1) d6(in=in2);
by bene_id;
if in1 and in2;
```

```
run;
******************************;
proc freq data=dd2 noprint;
table bene_id/out=dd3; run;

data dd4;
set dd3;
if count>=2;
keep bene_id;
run;

proc means data=dd2 noprint;
class bene_id ;
var DAYS_SUPLY_NUM;
output out=dd5 sum= ;
run;

data dd6;
set dd5;
if DAYS_SUPLY_NUM>60;
if bene_id ne '';
run;

data dd7;
merge dd4(in=in1) dd6(in=in2);
by bene_id;
if in1 and in2;
run;

********************************************************;

data d8;
set d6 dd6 ccw2.dementiadiag;
run;

proc sort data=d8 out=ccw2.dementia nodupkey;by bene_id;
run;

*****************numerator***********
ICD 9 codes
4:2950-2959
4:2960 2961 2964 - 2969
3334
30723****************************;

data dementia2diag;
set ccw2.d29;/*independent variable part for all diagnosis information*/
bene_id=hicno;
diag2=substr(diag,1,4);
if diag2='2950' or diag2='2951' or diag2='2952' or diag2='2953' or
diag2='2954' or diag2='2955' or diag2='2956' or diag2='2957'
or diag2='2958' or diag2='2959' or diag2='2960' or diag2='2961' or
diag2='2964' or diag2='2965' or diag2='2966' or diag2='2967'
or diag2='2968' or diag2='2969' or diag2='3334' or diag='30723' ;
keep bene_id diag;
run;
```

```
proc sort data=dementia2diag out=ccw2.dementia2diag nodupkey;
by bene_id; run;

******************************************;

proc import datafile="F:\jwang26\Part D data\psychotics.csv"
     out=psychotics
     dbms=csv
     replace;
     getnames=yes;
       guessingrows=32767;
run;
proc sort data=psychotics; by ndc;
run;

data ddd;
merge ccw2.pdcdrug(in=in1) psychotics(in=in2);
by ndc;
if in1 and in2;
run;
proc sort data=ddd out=ddd2 ; by bene_id  SRVC_DT;run;

proc means data=ddd2 noprint;
class bene_id ;
var DAYS_SUPLY_NUM;
output out=ddd3 sum= ;
run;

data ddd4;
set ddd3;
if DAYS_SUPLY_NUM>30;
if bene_id ne '';
run;

*******************;
data antipsychotics;
merge ccw2.dementia(in=in1) ddd4(in=in2) ccw2.dementia2diag(in=in3);
by bene_id;
if in1 and in2 and not in3;
antipsychotics=1;
keep bene_id antipsychotics;
run;

data ccw2.antipsychotics;
merge ccw2.dementia(in=in1)  antipsychotics;
by bene_id;
if in1;
if antipsychotics=. then antipsychotics=0;
keep bene_id antipsychotics;run;
```