

Title: Association Between Results of an Amyloid PET Scan and Health Care Utilization in Individuals with Cognitive Impairment

Software used for analyses: SAS Enterprise Guide 7.1 and Stata 17

Funding: This study was supported by a grant from National Institute on Aging (Grant # 1R01AG053934-01A1)

Syntax for: Merging crosswalk to VRDC data

Data Cleaning (SAS)

/*CARE IDEAS:

Program 1:

1. Merge crosswalk to finder file
 2. Find the unique 1-1 matches
- */

/*Merge finder file to crosswalk*/
/*Count of rows*/

*Read in Finder file, rename some columns;
data finder1;
set _uplds.carei_finder1_vrdc;
finder_sex = gender;
finder_dob = dob;
format finder_dob date9.;
drop gender dob;
run;

*Count unique Brown study ID #s;
proc sql;
create table num_ids_finder as select count (distinct brown_study_id) as num_brownid from finder1;
quit;
/*
Brown ID: N=2206
*/

*Count unique Brown ID #s and unique Bene ID #s;
proc sql;
create table num_ids_xwlk as select count (distinct unique_id) as num_brownid, count (distinct bene_id) as num_beneid
from IN055289.UNIQBENEX20_R9657;
quit;
/*
Brown ID: N=2206
Bene ID: N=2252
*/

*Join finder file and crosswalk file;
proc sql;
create table step1 as select * from finder1 as a left join IN055289.UNIQBENEX20_R9657 as b

```

on a.brown_study_id = b.unique_id;
quit;

/*N=2324 rows*/

/*Add in BSF info*/
data bsf_info;
set
IN055289.MBSF_ABCD15_R9657
IN055289.MBSF_ABCD16_R9657
IN055289.MBSF_ABCD17_R9657
IN055289.MBSF_ABCD18_R9657;
bsf_dob = bene_birth_dt;
if sex_ident_cd = 2 then bsf_sex = "Female";
else if sex_ident_cd = 1 then bsf_sex = "Male";
format bsf_dob date9.;
keep bene_id bsf_dob bsf_sex;
run;

proc sort data=bsf_info nodupkey;
by bene_id;
run;

proc sql;
create table step2 as select * from step1 as a left join bsf_info as b
on a.bene_id = b.bene_id;
quit;

/*Look for duplicate links*/
proc sort data=step2;
by unique_id;
run;

data dup1 nodup nobene;
set step2;
by unique_id;
if bene_id = . then output nobene;
else if first.unique_id and last.unique_id then output nodup;
else output dup1;
run;

proc sort data=nodup;
by bene_id;
run;

data nodup2 dup2;
set nodup;
by bene_id;
if first.bene_id and last.bene_id then output nodup2;
else output dup2;
run;

data dupall;
set dup1 dup2;

```

```

run;
proc sql;
create table numdups as select count (distinct unique_id) as num_dupid
from dupall;
quit;

/*
Match Summary:
no match: 29
1-to-1 match: N=2099
Not 1-1: N=196 rows, 78 IDs
*/

proc freq data=nodup2;
tables sex_match*dob_match;
run;

/*
Not matched DOB and SEX: 20
Matched DOB, not SEX: 7
Matched SEX, not DOB: 35
Matched both DOB and SEX: 2037
*/
data nodup2;
set nodup2;
if finder_dob = bsf_dob and finder_sex = bsf_sex then match = 1;
else match = 0;
run;
proc freq data = nodup2;
tables match;
run;

data final1;
set nodup2;
if match = 1 then output;
run;
/*Matched = 2035*/

/*
To do:
1. Look at list of duplicates and see if we can find the correct match
2. Look at those with mismatched info and see if there are any that look OK to use
(same gender, DOB a little off)
*/

*Look through duplicates for correct match - by matching DOB to bsf_dob;
*Reformat DOB variable;
data dupall;
set dupall;
format DOB date9.;
run;

*Find which rows have matching dob and sex between finder and bsf;
data matched_duplicates;
set dupall;

```

```
if finder_dob = bsf_dob and finder_sex = bsf_sex then output matched_duplicates;  
run;
```

```
*Eliminate rows that duplicate the unique_id variable;  
proc sort data=matched_duplicates nuniquekeys uniqueout = no_longer_dup out= still_dup;  
by unique_id;  
run;
```

```
*One example of duplicate bene_id - removed from data;  
data no_longer_dup;  
set no_longer_dup;  
if unique_id = 68450 then delete;  
run;
```

```
*Found 56 additional cases by correcting duplicates;
```

```
*2. Looking through mismatched info for usable data;
```

```
*Find rows with no duplicate that lack a match;  
data needs_match;  
set nodup2;  
if match = 0 then output needs_match;  
run;
```

```
*Reformat DOB;  
data needs_match;  
set needs_match;  
format DOB date9.;  
run;
```

```
*Eliminate rows that do not have a match on sex;  
data needs_match;  
set needs_match;  
if SEX_MATCH = 1 then output needs_match;  
run;
```

```
*Inspected remaining rows, extracted ones that have different DOB by a small amount, or missing bsf_dob;  
data matched;  
set needs_match;  
if _N_ in (1,2,3,5,6,8,12,13,14,15,16,17,19,20,22, 23, 24, 25, 26, 27, 28, 30, 31, 33,34,35,36,37) then output;  
run;
```

```
*Combine all data into one file: 2035 rows with no duplicates, 56 corrected duplicates, 28 matched rows;  
data final_patients;  
set final1 no_longer_dup matched;  
run;
```

```
*Count ID#s to make sure there are as many rows as ID numbers;  
proc sql;  
create table num_ids_tot as select count (distinct unique_id) as num_brownid, count (distinct bene_id) as  
num_beneid, count (distinct alzid) as num_alzid  
from no_longer_dup;  
quit;
```

```
data SH055289.Final_Patients;
```

```
set final_patients;  
run;
```

Syntax for: Merging patient data to survey data

```
/*Convert alzid to numeric in baseline survey data for merge with finder file */  
data baseline_survey_data_numeric_ID;  
set sh055289.baselinesurveydata;  
alzid = input(cpait_alzid, 6.);  
run;
```

```
/*Join all patients from finder file to baseline survey data*/  
proc sql;  
create table merged as select * from sh055289.final_patients as a left join baseline_survey_data_numeric_ID  
as b  
on a.alzid = b.alzid;  
quit;
```

```
proc freq data= merged;  
tables scanyear;  
run;
```

```
/*Use an anti-join to find all patients who were not joined*/  
proc sql;  
create table no_merge as select * from baseline_survey_data_numeric_ID t1 left join sh055289.final_patients  
t2  
on t1.alzid = t2.alzid  
where t2.alzid is NULL;  
quit;
```

```
/*Find frequency of demographics for joined patients*/  
proc freq data=merged;  
tables cpait_gender2 rec_cpait_race e_benif_type impairment_level af_amyloid_result REC_part_RELAT;  
run;
```

```
/*Find frequency of demographics for unjoined patients*/  
proc freq data=no_merge;  
tables cpait_gender2 rec_cpait_race e_benif_type impairment_level af_amyloid_result REC_part_RELAT;  
run;
```

```
/*Add a scan date variable which is the 15th of the scan month, in the scan year (no day is provided)*/  
data merged;  
set merged;  
scan_date = mdy(scanmonth, 15, scanyear);  
format scan_date date9.;  
run;  
data no_merge;  
set no_merge;  
scan_date = mdy(scanmonth, 15, scanyear);  
format scan_date date9.;  
run;
```

```
/*Find average age, scan month and scan year */  
proc means data=merged;  
var scanmonth scanyear REC_cpait_AGE;  
run;  
proc means data=no_merge;
```

```
var scanmonth scanyear REC_cpait_AGE;  
run;  
Syntax for: Determining utilization outcomes
```

```
/*Collect other utilizations (inpatient, outpatient, skilled nursing facility, home health*/
```

```
data ip_claims;  
set  
IN055289.INPTCLMSK15_R9657  
IN055289.INPTCLMSK16_R9657  
IN055289.INPTCLMSK17_R9657  
IN055289.INPTCLMSK18_R9657  
IN055289.INPTCLMSK19_R9657  
IN055289.INPTCLMSK20_R9990;  
if rev_cntr ^= "0450" or rev_cntr ^= "0451" or rev_cntr ^= "0452"  
or rev_cntr ^= "0453" or rev_cntr ^= "0454" or rev_cntr ^= "0455"  
or rev_cntr ^= "0456" or rev_cntr ^= "0457" or rev_cntr ^= "0458"  
or rev_cntr ^= "0459" or rev_cntr ^= "0981" then output ip_claims;  
keep bene_id clm_id clm_from_dt clm_thru_dt;  
run;
```

```
data op_claims;  
set  
IN055289.OTPTCLMSK15_R9657  
IN055289.OTPTCLMSK16_R9657  
IN055289.OTPTCLMSK17_R9657  
IN055289.OTPTCLMSK18_R9657  
IN055289.OTPTCLMSK19_R9657  
IN055289.OTPTCLMSK20_R9990;  
if rev_cntr ^= "0450" or rev_cntr ^= "0451" or rev_cntr ^= "0452"  
or rev_cntr ^= "0453" or rev_cntr ^= "0454" or rev_cntr ^= "0455"  
or rev_cntr ^= "0456" or rev_cntr ^= "0457" or rev_cntr ^= "0458"  
or rev_cntr ^= "0459" or rev_cntr ^= "0981" then output op_claims;  
keep bene_id clm_id clm_from_dt clm_thru_dt;  
run;
```

```
data op_rev;  
set  
IN055289.OTPTREVK15_R9657  
IN055289.OTPTREVK16_R9657  
IN055289.OTPTREVK17_R9657  
IN055289.OTPTREVK18_R9657  
IN055289.OTPTREVK19_R9657  
IN055289.OTPTREVK20_R9990;  
if rev_cntr ^= "0450" and rev_cntr ^= "0451" and rev_cntr ^= "0452"  
and rev_cntr ^= "0453" and rev_cntr ^= "0454" and rev_cntr ^= "0455"  
and rev_cntr ^= "0456" and rev_cntr ^= "0457" and rev_cntr ^= "0458"  
and rev_cntr ^= "0459" and rev_cntr ^= "0981" then output op_rev;  
keep bene_id clm_id clm_from_dt clm_thru_dt;  
run;
```

```
data ip_rev;  
set  
IN055289.INPTREVK15_R9657  
IN055289.INPTREVK16_R9657  
IN055289.INPTREVK17_R9657  
IN055289.INPTREVK18_R9657
```

```

IN055289.INPTREVK19_R9657
IN055289.INPTREVK20_R9990;
if rev_cntr ^= "0450" and rev_cntr ^= "0451" and rev_cntr ^= "0452"
and rev_cntr ^= "0453" and rev_cntr ^= "0454" and rev_cntr ^= "0455"
and rev_cntr ^= "0456" and rev_cntr ^= "0457" and rev_cntr ^= "0458"
and rev_cntr ^= "0459" and rev_cntr ^= "0981" then output ip_rev;
keep bene_id clm_id clm_from_dt clm_thru_dt;
run;

proc sql;
create table ip_claims as select * from ip_rev as a left join ip_claims as b
on a.clm_id = b.clm_id;
quit;

proc sql;
create table op_claims as select * from op_rev as a left join op_claims as b
on a.clm_id = b.clm_id;
quit;

data snf_claims;
set
IN055289.SNFCLMSK15_R9657
IN055289.SNFCLMSK16_R9657
IN055289.SNFCLMSK17_R9657
IN055289.SNFCLMSK18_R9657
IN055289.SNFCLMSK19_R9657
IN055289.SNFCLMSK20_R9990;
keep bene_id clm_id clm_from_dt;
run;

data hha_claims;
set
IN055289.HHACLMSK15_R9657
IN055289.HHACLMSK16_R9657
IN055289.HHACLMSK17_R9657
IN055289.HHACLMSK18_R9657
IN055289.HHACLMSK19_R9657
IN055289.HHACLMSK20_R9990;
keep bene_id clm_id clm_from_dt;
run;

/*Remove duplicates from each of the utilization categories*/

proc sort data=ip_claims nodupkey dupout=ip_dup;
by bene_id clm_from_dt clm_thru_dt;
run;

proc sort data=op_claims nodupkey dupout=op_dup;
by bene_id clm_from_dt clm_thru_dt;
run;

proc sort data=snf_claims nodupkey;
by bene_id clm_from_dt;
run;

proc sort data=hha_claims nodupkey;

```

```

by clm_id;
run;

/*Recode months to 1-63*/
data ip_claims;
set ip_claims;
month = put(clm_from_dt, Month2.);
year = put(clm_from_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data op_claims;
set op_claims;
month = put(clm_from_dt, Month2.);
year = put(clm_from_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data snf_claims;
set snf_claims;
month = put(clm_from_dt, Month2.);
year = put(clm_from_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data hha_claims;
set hha_claims;
month = put(clm_from_dt, Month2.);
year = put(clm_from_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

/*Convert IP claims to IP1-IP63 with how many IP claims a patient had in each of the 60 months*/
proc sort data=ip_claims;
by bene_id;
run;

proc transpose data=ip_claims out=wide prefix=ip_mo_num;
by bene_id;
var mo_number;
run;

data ip_data;
set wide;
array ip_monums (32) ip_mo_num1-ip_mo_num32;
array IPs (63) IP1 - IP63;
do i = 1 to 63;
    IPs(i) = 0;
    do j = 1 to 32;
        if ip_monums(j) = i then IPs(i) = IPs(i) + 1;
    end;
end;
run;

/*Convert OP claims to OP1-OP63 with how many OP claims a patient had in each of the 63 months*/

```



```
proc sort data=op_claims;
by bene_id;
run;
```

```
proc transpose data=op_claims out=wide prefix=op_mo_num;
by bene_id;
var mo_number;
run;
```

```
data op_data;
set wide;
array op_monums (92) op_mo_num1-op_mo_num92;
array OPs (63) OP1 - OP63;
do i = 1 to 63;
    OPs(i) = 0;
    do j = 1 to 92;
        if op_monums(j) = i then OPs(i) = OPs(i) + 1;
    end;
end;
run;
```

```
/*Convert SNF claims to SNF1-SNF60 with how many IP claims a patient had in each of the 60 months*/
proc sort data=snf_claims;
by bene_id;
run;
```

```
proc transpose data=snf_claims out=wide prefix=snf_mo_num;
by bene_id;
var mo_number;
run;
```

```
data snf_data;
set wide;
array snf_monums (20) snf_mo_num1-snf_mo_num20;
array SNFs (63) SNF1 - SNF63;
do i = 1 to 63;
    SNFs(i) = 0;
    do j = 1 to 20;
        if snf_monums(j) = i then SNFs(i) = SNFs(i) + 1;
    end;
end;
run;
```

```
/*Convert HHA claims to HHA1-HHA60 with how many HHA claims a patient had in each of the 63 months*/
proc sort data=hha_claims;
by bene_id;
run;
```

```
proc transpose data=hha_claims out=wide prefix=hha_mo_num;
by bene_id;
var mo_number;
run;
```

```
data hha_data;
```

```

set wide;
array hha_monums (32) hha_mo_num1-hha_mo_num32;
array HHAs (63) HHA1 - HHA63;
do i = 1 to 63;
    HHAs(i) = 0;
    do j = 1 to 32;
        if hha_monums(j) = i then HHAs(i) = HHAs(i) + 1;
    end;
end;
end;
run;

```

```

/*Merge to HMO/Dual status by month tables, and change missing values to 0 in utilization columns*/
proc sql;
create table merged_utils as select * from sh055289.er_by_month as a left join ip_data as b
on a.bene_id = b.bene_id;
quit;

```

```

proc sql;
create table merged_utils as select * from merged_utils as a left join op_data as b
on a.bene_id = b.bene_id;
quit;

```

```

proc sql;
create table merged_utils as select * from merged_utils as a left join snf_data as b
on a.bene_id = b.bene_id;
quit;

```

```

proc sql;
create table merged_utils as select * from merged_utils as a left join hha_data as b
on a.bene_id = b.bene_id;
quit;

```

```

data merged_utils;
set merged_utils;
array IPs (63) IP1 - IP63;
array OPs (63) OP1 - OP63;
array SNFs (63) SNF1 - SNF63;
array HHAs (63) HHA1 - HHA63;
do i = 1 to 63;
    if IPs(i) = . then IPs(i) = 0;
    if OPs(i) = . then OPs(i) = 0;
    if SNFs(i) = . then SNFs(i) = 0;
    if HHAs(i) = . then HHAs(i) = 0;
end;
run;

```

```

/*Convert to month by month relative to scan month (newX1 is 18 months pre scan, newX43 is 24 months post scan*/
data monthly_data;
set merged_utils;
array newIP (43) newIP1 - newIP43;
array IP (63) IP1-IP63;
array newOP (43) newOP1 - newOP43;
array OP (63) OP1-OP63;
array newSNF (43) newSNF1 - newSNF43;

```

```

array SNF (63) SNF1-SNF63;
array newHHA (43) newHHA1 - newHHA43;
array HHA (63) HHA1-HHA63;
do i = 1 to 43;
if (i-1) + new_pre_scan < 1 then newIP(i) = .;
else if (i-1) + new_pre_scan > 63 then newIP(i) = .;
else newIP(i) = IP((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newOP(i) = .;
else if (i-1) + new_pre_scan > 63 then newOP(i) = .;
else newOP(i) = OP((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newSNF(i) = .;
else if (i-1) + new_pre_scan > 63 then newSNF(i) = .;
else newSNF(i) = SNF((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newHHA(i) = .;
else if (i-1) + new_pre_scan > 63 then newHHA(i) = .;
else newHHA(i) = HHA((i-1) + new_pre_scan);
end;
run;

```

```

data monthly_data;
set monthly_data;
keep bene_id scanyear scanmonth new_scan_month nHMO1-nHMO63 nDUAL1-nDUAL63 ER1-ER63
IP1-IP63 OP1-OP63 SNF1-SNF63 HHA1-HHA63
newHMO1 - newHMO43 newdual1 - newdual43 newER1 - newER43 newIP1 - newIP43
newOP1-newOP43 newSNF1 - newSNF43 newHHA1-newHHA43;
run;

```

```

/*Merge monthly utilization data to survey results/MBSF file*/
proc sql;
create table care as select * from sh055289.merged as a left join sh055289.monthly_data as b
on a.bene_id = b.bene_id;
quit;

```

```

data care2;
set care;
keep bene_id scan_date af_amyloid_result cpait_RsltPosNeg REC_cpait_CrrctIntrtScn_Bnry
REC_cpait_CrrctIntrtScn_Cat
newHMO1-newHMO43 newdual1-newdual43 newER1-newER43 newIP1-newIP43 newOP1-newOP43
newSNF1-newSNF43 newHHA1-newHHA43;
run;

```

```

/*Convert monthly utilizations to quarterly utilizations*/
data care_er;
set care2;
ERQ1 = newER1 + newER2 + newER3;
ERQ2 = newER4 + newER5 + newER6;
ERQ3 = newER7 + newER8 + newER9;
ERQ4 = newER10 + newER11 + newER12;
ERQ5 = newER13 + newER14 + newER15;
ERQ6 = newER16 + newER17 + newER18;
ERQ7 = newER20 + newER21 + newER22;
ERQ8 = newER23 + newER24 + newER25;
ERQ9 = newER26 + newER27 + newER28;
ERQ10 = newER29 + newER30 + newER31;
ERQ11 = newER32 + newER33 + newER34;
ERQ12 = newER35 + newER36 + newER37;

```

ERQ13 = newER38 + newER39 + newER40;
 ERQ14 = newER41 + newER42 + newER43;
 IPQ1 = newIP1 + newIP2 + newIP3;
 IPQ2 = newIP4 + newIP5 + newIP6;
 IPQ3 = newIP7 + newIP8 + newIP9;
 IPQ4 = newIP10 + newIP11 + newIP12;
 IPQ5 = newIP13 + newIP14 + newIP15;
 IPQ6 = newIP16 + newIP17 + newIP18;
 IPQ7 = newIP20 + newIP21 + newIP22;
 IPQ8 = newIP23 + newIP24 + newIP25;
 IPQ9 = newIP26 + newIP27 + newIP28;
 IPQ10 = newIP29 + newIP30 + newIP31;
 IPQ11 = newIP32 + newIP33 + newIP34;
 IPQ12 = newIP35 + newIP36 + newIP37;
 IPQ13 = newIP38 + newIP39 + newIP40;
 IPQ14 = newIP41 + newIP42 + newIP43;
 OPQ1 = newOP1 + newOP2 + newOP3;
 OPQ2 = newOP4 + newOP5 + newOP6;
 OPQ3 = newOP7 + newOP8 + newOP9;
 OPQ4 = newOP10 + newOP11 + newOP12;
 OPQ5 = newOP13 + newOP14 + newOP15;
 OPQ6 = newOP16 + newOP17 + newOP18;
 OPQ7 = newOP20 + newOP21 + newOP22;
 OPQ8 = newOP23 + newOP24 + newOP25;
 OPQ9 = newOP26 + newOP27 + newOP28;
 OPQ10 = newOP29 + newOP30 + newOP31;
 OPQ11 = newOP32 + newOP33 + newOP34;
 OPQ12 = newOP35 + newOP36 + newOP37;
 OPQ13 = newOP38 + newOP39 + newOP40;
 OPQ14 = newOP41 + newOP42 + newOP43;
 SNFQ1 = newSNF1 + newSNF2 + newSNF3;
 SNFQ2 = newSNF4 + newSNF5 + newSNF6;
 SNFQ3 = newSNF7 + newSNF8 + newSNF9;
 SNFQ4 = newSNF10 + newSNF11 + newSNF12;
 SNFQ5 = newSNF13 + newSNF14 + newSNF15;
 SNFQ6 = newSNF16 + newSNF17 + newSNF18;
 SNFQ7 = newSNF20 + newSNF21 + newSNF22;
 SNFQ8 = newSNF23 + newSNF24 + newSNF25;
 SNFQ9 = newSNF26 + newSNF27 + newSNF28;
 SNFQ10 = newSNF29 + newSNF30 + newSNF31;
 SNFQ11 = newSNF32 + newSNF33 + newSNF34;
 SNFQ12 = newSNF35 + newSNF36 + newSNF37;
 SNFQ13 = newSNF38 + newSNF39 + newSNF40;
 SNFQ14 = newSNF41 + newSNF42 + newSNF43;
 HHAQ1 = newHHA1 + newHHA2 + newHHA3;
 HHAQ2 = newHHA4 + newHHA5 + newHHA6;
 HHAQ3 = newHHA7 + newHHA8 + newHHA9;
 HHAQ4 = newHHA10 + newHHA11 + newHHA12;
 HHAQ5 = newHHA13 + newHHA14 + newHHA15;
 HHAQ6 = newHHA16 + newHHA17 + newHHA18;
 HHAQ7 = newHHA20 + newHHA21 + newHHA22;
 HHAQ8 = newHHA23 + newHHA24 + newHHA25;
 HHAQ9 = newHHA26 + newHHA27 + newHHA28;
 HHAQ10 = newHHA29 + newHHA30 + newHHA31;
 HHAQ11 = newHHA32 + newHHA33 + newHHA34;
 HHAQ12 = newHHA35 + newHHA36 + newHHA37;

```
HHAQ13 = newHHA38 + newHHA39 + newHHA40;
HHAQ14 = newHHA41 + newHHA42 + newHHA43;
run;
```

```
/*Find average utilizations, stratified by AF_AMYLOID_RESULT (scan result)*/
proc sort data=care_er;
by af_amyloid_result;
run;
```

```
proc means data=care_er;
var ERQ3 -ERQ14 IPQ3 - IPQ14 OPQ3-OPQ14 SNFQ3 -SNFQ14 HHAQ3 - HHAQ14;
class af_amyloid_result;
output out=util_means1;
run;
```

```
data util_means1;
set util_means1;
if _STAT_ = "MEAN" and (af_amyloid_result = 1 or af_amyloid_result = 2) then output util_means1;
keep af_amyloid_result ERQ3-ERQ14 IPQ3 - IPQ14 OPQ3-OPQ14 SNFQ3 -SNFQ14 HHAQ3 - HHAQ14;
run;
```

```
/*Find average utilizations, stratified by REC_cpait_CrrctIntrtScn_Cat (scan result vs. patient's interpretation of
scan result)*/
proc sort data=care_er;
by REC_cpait_CrrctIntrtScn_Cat;
run;
```

```
proc means data=care_er;
var ERQ3 -ERQ14 IPQ3 - IPQ14 OPQ3-OPQ14 SNFQ3 -SNFQ14 HHAQ3 - HHAQ14;
class REC_cpait_CrrctIntrtScn_Cat;
output out=util_means2;
run;
```

```
data util_means2;
set util_means2;
if _STAT_ = "MEAN" and (rec_cpait_crrctintrtscn_cat = 1 or rec_cpait_crrctintrtscn_cat = 2 or
rec_cpait_crrctintrtscn_cat = 3 or rec_cpait_crrctintrtscn_cat = 4) then output util_means2;
keep REC_cpait_CrrctIntrtScn_Cat ERQ3-ERQ14 IPQ3 - IPQ14 OPQ3-OPQ14 SNFQ3 -SNFQ14 HHAQ3 -
HHAQ14;
run;
```

```
data quarterly_utilizations;
set care_er;
keep bene_id af_amyloid_result REC_cpait_CrrctIntrtScn_Cat ERQ3-ERQ14 IPQ3 - IPQ14 OPQ3-OPQ14
SNFQ3 -SNFQ14 HHAQ3 - HHAQ14;
run;
```

```
data quarterly_utilizations;
set quarterly_utilizations (rename=(ERQ3-ERQ14 = ERQ1 - ERQ12 IPQ3-IPQ14 = IPQ1-IPQ12 OPQ3-OPQ14
= OPQ1-OPQ12 SNFQ3-SNFQ14 = SNFQ1 - SNFQ12 HHAQ3-HHAQ14 = HHAQ1-HHAQ12));
run;
```

```
/*Collect nervous system test related claims*/
data nervous_system_codes;
set in055289.bcarlinek15_r9657
in055289.bcarlinek16_r9657
```

```

in055289.bcarlinek17_r9657
in055289.bcarlinek18_r9657
in055289.bcarlinek19_r9657
in055289.bcarlinek20_r9990
in055289.bcarlinek20_r10185
in055289.bcarlinek20_r10440;
if hcpcs_cd ~= "96116" and hcpcs_cd ~= "96118" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data nervous_system_codes_2;
set in055289.otptrevk15_r9657
in055289.otptrevk16_r9657
in055289.otptrevk17_r9657
in055289.otptrevk18_r9657
in055289.otptrevk19_r9657
in055289.otptrevk20_r9990
in055289.otptrevk20_r10185
in055289.otptrevk20_r10440;
if hcpcs_cd ~= "96116" and hcpcs_cd ~= "96118" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data nervous_system_codes;
set nervous_system_codes
nervous_system_codes_2;
run;

```

```

proc sort data=nervous_system_codes nodupkeys;
by clm_id hcpcs_cd;
run;

```

```

/*Collect psychotherapy related claims*/
data psychotherapy_codes;
set in055289.bcarlinek15_r9657
in055289.bcarlinek16_r9657
in055289.bcarlinek17_r9657
in055289.bcarlinek18_r9657
in055289.bcarlinek19_r9657
in055289.bcarlinek20_r9990
in055289.bcarlinek20_r10185
in055289.bcarlinek20_r10440;
if hcpcs_cd ~= "90832" and hcpcs_cd ~= "90834" and hcpcs_cd ~= "90837" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data psychotherapy_codes_2;
set in055289.otptrevk15_r9657
in055289.otptrevk16_r9657
in055289.otptrevk17_r9657
in055289.otptrevk18_r9657
in055289.otptrevk19_r9657
in055289.otptrevk20_r9990
in055289.otptrevk20_r10185
in055289.otptrevk20_r10440;
if hcpcs_cd ~= "90832" and hcpcs_cd ~= "90834" and hcpcs_cd ~= "90837" then delete;

```

```
keep bene_id clm_id hcpcs_cd clm_thru_dt;  
run;
```

```
data psychotherapy_codes;  
set psychotherapy_codes  
psychotherapy_codes_2;  
run;
```

```
proc sort data=psychotherapy_codes nodupkeys;  
by clm_id hcpcs_cd;  
run;
```

```
/*Collect other nervous system test related claims*/  
data other_nervous_codes;  
set in055289.bcarlinek15_r9657  
in055289.bcarlinek16_r9657  
in055289.bcarlinek17_r9657  
in055289.bcarlinek18_r9657  
in055289.bcarlinek19_r9657  
in055289.bcarlinek20_r9990  
in055289.bcarlinek20_r10185  
in055289.bcarlinek20_r10440;  
if hcpcs_cd ~= "96150" and hcpcs_cd ~= "96152" and hcpcs_cd ~= "96154" then delete;  
keep bene_id clm_id hcpcs_cd clm_thru_dt;  
run;
```

```
data other_nervous_codes_2;  
set in055289.otptrevk15_r9657  
in055289.otptrevk16_r9657  
in055289.otptrevk17_r9657  
in055289.otptrevk18_r9657  
in055289.otptrevk19_r9657  
in055289.otptrevk20_r9990  
in055289.otptrevk20_r10185  
in055289.otptrevk20_r10440;  
if hcpcs_cd ~= "96150" and hcpcs_cd ~= "96152" and hcpcs_cd ~= "96154" then delete;  
keep bene_id clm_id hcpcs_cd clm_thru_dt;  
run;
```

```
data other_nervous_codes;  
set other_nervous_codes  
other_nervous_codes_2;  
run;
```

```
proc sort data=other_nervous_codes nodupkeys;  
by clm_id hcpcs_cd;  
run;
```

```
/*Collect diagnostic test related claims*/  
data diagnostic_codes;  
set in055289.bcarlinek15_r9657  
in055289.bcarlinek16_r9657  
in055289.bcarlinek17_r9657  
in055289.bcarlinek18_r9657  
in055289.bcarlinek19_r9657  
in055289.bcarlinek20_r9990
```

```

in055289.bcarlinek20_r10185
in055289.bcarlinek20_r10440;
if hcpcs_cd ~= "70551" and hcpcs_cd ~= "70553" and hcpcs_cd ~= "78608"
and hcpcs_cd ~= "78609" and hcpcs_cd ~= "78607" and hcpcs_cd ~= "62270"
and hcpcs_cd ~= "93880" and hcpcs_cd ~= "93882" and hcpcs_cd ~= "93886"
and hcpcs_cd ~= "93888" and hcpcs_cd ~= "95816" and hcpcs_cd ~= "95819" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data diagnostic_codes_2;
set in055289.otptrevk15_r9657
in055289.otptrevk16_r9657
in055289.otptrevk17_r9657
in055289.otptrevk18_r9657
in055289.otptrevk19_r9657
in055289.otptrevk20_r9990
in055289.otptrevk20_r10185
in055289.otptrevk20_r10440;
if hcpcs_cd ~= "70551" and hcpcs_cd ~= "70553" and hcpcs_cd ~= "78608"
and hcpcs_cd ~= "78609" and hcpcs_cd ~= "78607" and hcpcs_cd ~= "62270"
and hcpcs_cd ~= "93880" and hcpcs_cd ~= "93882" and hcpcs_cd ~= "93886"
and hcpcs_cd ~= "93888" and hcpcs_cd ~= "95816" and hcpcs_cd ~= "95819" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data diagnostic_codes;
set diagnostic_codes
diagnostic_codes_2;
run;

```

```

proc sort data=diagnostic_codes nodupkeys;
by clm_id hcpcs_cd;
run;

```

```

/*Collect other diagnostic test related claims*/
data other_diagnostic_codes;
set in055289.bcarlinek15_r9657
in055289.bcarlinek16_r9657
in055289.bcarlinek17_r9657
in055289.bcarlinek18_r9657
in055289.bcarlinek19_r9657
in055289.bcarlinek20_r9990
in055289.bcarlinek20_r10185
in055289.bcarlinek20_r10440;
if hcpcs_cd ~= "83520" and hcpcs_cd ~= "86403" and hcpcs_cd ~= "86592"
and hcpcs_cd ~= "86780" and hcpcs_cd ~= "81401" and hcpcs_cd ~= "81405"
and hcpcs_cd ~= "81406" and hcpcs_cd ~= "81479" and hcpcs_cd ~= "84999"
and hcpcs_cd ~= "83516" and hcpcs_cd ~= "83519" and hcpcs_cd ~= "84181"
and hcpcs_cd ~= "84182" and hcpcs_cd ~= "86255" and hcpcs_cd ~= "86376"
and hcpcs_cd ~= "86800" and hcpcs_cd ~= "86701" and hcpcs_cd ~= "86702"
and hcpcs_cd ~= "86703" and hcpcs_cd ~= "82310" and hcpcs_cd ~= "82397"
and hcpcs_cd ~= "83519" and hcpcs_cd ~= "83520" and hcpcs_cd ~= "86255"
and hcpcs_cd ~= "86256" and hcpcs_cd ~= "84182" and hcpcs_cd ~= "86341"
then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```



```

data other_diagnostic_codes_2;
set in055289.otptrevk15_r9657
in055289.otptrevk16_r9657
in055289.otptrevk17_r9657
in055289.otptrevk18_r9657
in055289.otptrevk19_r9657
in055289.otptrevk20_r9990
in055289.otptrevk20_r10185
in055289.otptrevk20_r10440;
if hcpcs_cd ~= "83520" and hcpcs_cd ~= "86403" and hcpcs_cd ~= "86592"
and hcpcs_cd ~= "86780" and hcpcs_cd ~= "81401" and hcpcs_cd ~= "81405"
and hcpcs_cd ~= "81406" and hcpcs_cd ~= "81479" and hcpcs_cd ~= "84999"
and hcpcs_cd ~= "83516" and hcpcs_cd ~= "83519" and hcpcs_cd ~= "84181"
and hcpcs_cd ~= "84182" and hcpcs_cd ~= "86255" and hcpcs_cd ~= "86376"
and hcpcs_cd ~= "86800" and hcpcs_cd ~= "86701" and hcpcs_cd ~= "86702"
and hcpcs_cd ~= "86703" and hcpcs_cd ~= "82310" and hcpcs_cd ~= "82397"
and hcpcs_cd ~= "83519" and hcpcs_cd ~= "83520" and hcpcs_cd ~= "86255"
and hcpcs_cd ~= "86256" and hcpcs_cd ~= "84182" and hcpcs_cd ~= "86341" then delete;
keep bene_id clm_id hcpcs_cd clm_thru_dt;
run;

```

```

data other_diagnostic_codes;
set other_diagnostic_codes
other_diagnostic_codes_2;
run;

```

```

proc sort data=other_diagnostic_codes nodupkeys;
by clm_id hcpcs_cd;
run;

```

```

/*Look through carrier files to make sure claims are not the amyloid scan*/
data carrier_files;
set in055289.bcarlinek15_r9657
in055289.bcarlinek16_r9657
in055289.bcarlinek17_r9657
in055289.bcarlinek18_r9657
in055289.bcarlinek19_r9657
in055289.bcarlinek20_r9990
in055289.bcarlinek20_r10185;
keep bene_id clm_id clm_thru_dt prvdr_spclty betos_cd hcpcs_cd;
run;

```

```

data carrier_files;
set carrier_files;
if hcpcs_cd ~= "78811" and hcpcs_cd ~= "78814" then delete;
run;

```

```

/*Exclude the amyloid scan from the list of claims in each category*/
proc sql;
create table nervous_system_codes as select * from nervous_system_codes where clm_id not in (select
clm_id from carrier_files);
quit;

proc sql;

```

```

create table psychotherapy_codes as select * from psychotherapy_codes where clm_id not in (select clm_id
from carrier_files);
quit;

proc sql;
create table other_nervous_codes as select * from other_nervous_codes where clm_id not in (select clm_id
from carrier_files);
quit;

proc sql;
create table diagnosis_codes as select * from diagnosis_codes where clm_id not in (select clm_id from
carrier_files);
quit;

proc sql;
create table other_diagnostic_codes as select * from other_diagnostic_codes where clm_id not in (select
clm_id from carrier_files);
quit;

/*Recode months to 1-63*/
data nervous_system_codes;
set nervous_system_codes;
month = put(clm_thru_dt, Month2.);
year = put(clm_thru_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data psychotherapy_codes;
set psychotherapy_codes;
month = put(clm_thru_dt, Month2.);
year = put(clm_thru_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data other_nervous_codes;
set other_nervous_codes;
month = put(clm_thru_dt, Month2.);
year = put(clm_thru_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data diagnostic_codes;
set diagnostic_codes;
month = put(clm_thru_dt, Month2.);
year = put(clm_thru_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

data other_diagnostic_codes;
set other_diagnostic_codes;
month = put(clm_thru_dt, Month2.);
year = put(clm_thru_dt, Year4.);
mo_number = (12 * (year - 2015)) + month;
run;

```

```

/*Convert nervous system claims to NSC1-NSC63 with how many NSC claims a patient had in each of the 63
months*/
proc sort data=nervous_system_codes;
by bene_id;
run;

proc transpose data=nervous_system_codes out=wide prefix=NSC_mo_num;
by bene_id;
var mo_number;
run;

data NSC_data;
set wide;
array NSC_monums (39) NSC_mo_num1-NSC_mo_num39;
array NSCs (63) NSC1 - NSC63;
do i = 1 to 63;
    NSCs(i) = 0;
    do j = 1 to 39;
        if NSC_monums(j) = i then NSCs(i) = NSCs(i) + 1;
    end;
end;
run;

/*Convert psychotherapy claims to PT1-PT63 with how many PT claims a patient had in each of the 63
months*/
proc sort data=psychotherapy_codes;
by bene_id;
run;

proc transpose data=psychotherapy_codes out=wide prefix=pt_mo_num;
by bene_id;
var mo_number;
run;

data pt_data;
set wide;
array pt_monums (271) pt_mo_num1-pt_mo_num271;
array PTs (63) PT1 - PT63;
do i = 1 to 63;
    PTs(i) = 0;
    do j = 1 to 271;
        if pt_monums(j) = i then PTs(i) = PTs(i) + 1;
    end;
end;
run;

/*Convert other nervous system claims to ONC1-ONC63 with how many ONC claims a patient had in each of
the 63 months*/
proc sort data=other_nervous_codes;
by bene_id;
run;

proc transpose data=other_nervous_codes out=wide prefix=onc_mo_num;
by bene_id;

```

```
var mo_number;  
run;
```

```
data ONC_data;  
set wide;  
array ONC_monums (68) ONC_mo_num1-ONC_mo_num68;  
array ONCs (63) ONC1 - ONC63;  
do i = 1 to 63;  
    ONCs(i) = 0;  
    do j = 1 to 68;  
        if ONC_monums(j) = i then ONCs(i) = ONCs(i) + 1;  
    end;  
end;  
run;
```

```
/*Convert diagnostic claims to DC1-DC63 with how many DC claims a patient had in each of the 63 months*/  
proc sort data=diagnostic_codes;  
by bene_id;  
run;
```

```
proc transpose data=diagnostic_codes out=wide prefix=dc_mo_num;  
by bene_id;  
var mo_number;  
run;
```

```
data DC_data;  
set wide;  
array DC_monums (26) DC_mo_num1-DC_mo_num26;  
array DCs (63) DC1 - DC63;  
do i = 1 to 63;  
    DCs(i) = 0;  
    do j = 1 to 26;  
        if DC_monums(j) = i then DCs(i) = DCs(i) + 1;  
    end;  
end;  
run;
```

```
/*Convert other diagnostic claims to ODC1-ODC63 with how many ODC claims a patient had in each of the 63 months*/  
proc sort data=other_diagnostic_codes;  
by bene_id;  
run;
```

```
proc transpose data=other_diagnostic_codes out=wide prefix=odc_mo_num;  
by bene_id;  
var mo_number;  
run;
```

```
data ODC_data;  
set wide;  
array ODC_monums (43) ODC_mo_num1-ODC_mo_num43;  
array ODCs (63) ODC1 - ODC63;  
do i = 1 to 63;  
    ODCs(i) = 0;  
    do j = 1 to 43;  
        if ODC_monums(j) = i then ODCs(i) = ODCs(i) + 1;  
    end;  
end;  
run;
```

```

        if ODC_monums(j) = i then ODCs(i) = ODCs(i) + 1;
    end;
end;
run;

data scan_dates;
set sh055289.er_by_month;
keep bene_id new_scan_month new_post_scan new_pre_scan;
run;

proc sql;
create table ffs_scan_data as select * from sh055289.ffs_patients as a left join scan_dates as b
on a.bene_id = b.bene_id;
quit;

/*Merge to HMO/Dual status by month tables, and change missing values to 0 in utilization columns*/
proc sql;
create table merged_codes as select * from sh055289.er_by_month as a left join NSC_data as b
on a.bene_id = b.bene_id;
quit;

proc sql;
create table merged_codes as select * from merged_codes as a left join PT_data as b
on a.bene_id = b.bene_id;
quit;

proc sql;
create table merged_codes as select * from merged_codes as a left join ONC_data as b
on a.bene_id = b.bene_id;
quit;

proc sql;
create table merged_codes as select * from merged_codes as a left join DC_data as b
on a.bene_id = b.bene_id;
quit;

proc sql;
create table merged_codes as select * from merged_codes as a left join ODC_data as b
on a.bene_id = b.bene_id;
quit;

data merged_codes;
set merged_codes;
array NSCs (63) NSC1 - NSC63;
array PTs (63) PT1 - PT63;
array ONCs (63) ONC1 - ONC63;
array DCs (63) DC1 - DC63;
array ODCs (63) ODC1 - ODC63;
do i = 1 to 63;
if NSCs(i) = . then NSCs(i) = 0;
if PTs(i) = . then PTs(i) = 0;
if ONCs(i) = . then ONCs(i) = 0;
if DCs(i) = . then DCs(i) = 0;
if ODCs(i) = . then ODCs(i) = 0;
end;
run;

```

```

/*Convert to month by month relative to scan month (newX1 is 18 months pre scan, newX43 is 24 months post scan*/
data monthly_data;
set merged_codes;
array newNSC (43) newNSC1 - newNSC43;
array NSC (63) NSC1-NSC63;
array newPT (43) newPT1 - newPT43;
array PT (63) PT1-PT63;
array newONC (43) newONC1 - newONC43;
array ONC (63) ONC1-ONC63;
array newDC (43) newDC1 - newDC43;
array DC (63) DC1-DC63;
array newODC (43) newODC1 - newODC43;
array ODC (63) ODC1-ODC63;
do i = 1 to 43;
if (i-1) + new_pre_scan < 1 then newNSC(i) = .;
else if (i-1) + new_pre_scan > 63 then newNSC(i) = .;
else newNSC(i) = NSC((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newPT(i) = .;
else if (i-1) + new_pre_scan > 63 then newPT(i) = .;
else newPT(i) = PT((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newONC(i) = .;
else if (i-1) + new_pre_scan > 63 then newONC(i) = .;
else newONC(i) = ONC((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newDC(i) = .;
else if (i-1) + new_pre_scan > 63 then newDC(i) = .;
else newDC(i) = DC((i-1) + new_pre_scan);
if (i-1) + new_pre_scan < 1 then newODC(i) = .;
else if (i-1) + new_pre_scan > 63 then newODC(i) = .;
else newODC(i) = ODC((i-1) + new_pre_scan);
end;
run;

data monthly_data;
set monthly_data;
keep bene_id scanyear scanmonth new_scan_month nHMO1-nHMO63 nDUAL1-nDUAL63 NSC1-NSC63
PT1-PT63 ONC1-ONC63 DC1-DC63 ODC1-ODC63
newHMO1 - newHMO43 newdual1 - newdual43 newNSC1 - newNSC43 newPT1 - newPT43
newONC1-newONC43 newDC1 - newDC43 newODC1-newODC43;
run;

/*Merge monthly utilization data to survey results/MBSF file*/
proc sql;
create table care as select * from sh055289.merged as a left join monthly_data as b
on a.bene_id = b.bene_id;
quit;

data care2;
set care;
keep bene_id scan_date af_amyloid_result cpait_RsltPosNeg REC_cpait_CrrctIntrtScn_Bnry
REC_cpait_CrrctIntrtScn_Cat
newHMO1-newHMO43 newdual1-newdual43 newNSC1 - newNSC43 newPT1 - newPT43
newONC1-newONC43 newDC1 - newDC43 newODC1-newODC43;
run;

```

/*Convert monthly utilizations to quarterly utilizations*/

data care_codes;

set care2;

NSCQ1 = newNSC1 + newNSC2 + newNSC3;

NSCQ2 = newNSC4 + newNSC5 + newNSC6;

NSCQ3 = newNSC7 + newNSC8 + newNSC9;

NSCQ4 = newNSC10 + newNSC11 + newNSC12;

NSCQ5 = newNSC13 + newNSC14 + newNSC15;

NSCQ6 = newNSC16 + newNSC17 + newNSC18;

NSCQ7 = newNSC20 + newNSC21 + newNSC22;

NSCQ8 = newNSC23 + newNSC24 + newNSC25;

NSCQ9 = newNSC26 + newNSC27 + newNSC28;

NSCQ10 = newNSC29 + newNSC30 + newNSC31;

NSCQ11 = newNSC32 + newNSC33 + newNSC34;

NSCQ12 = newNSC35 + newNSC36 + newNSC37;

NSCQ13 = newNSC38 + newNSC39 + newNSC40;

NSCQ14 = newNSC41 + newNSC42 + newNSC43;

PTQ1 = newPT1 + newPT2 + newPT3;

PTQ2 = newPT4 + newPT5 + newPT6;

PTQ3 = newPT7 + newPT8 + newPT9;

PTQ4 = newPT10 + newPT11 + newPT12;

PTQ5 = newPT13 + newPT14 + newPT15;

PTQ6 = newPT16 + newPT17 + newPT18;

PTQ7 = newPT20 + newPT21 + newPT22;

PTQ8 = newPT23 + newPT24 + newPT25;

PTQ9 = newPT26 + newPT27 + newPT28;

PTQ10 = newPT29 + newPT30 + newPT31;

PTQ11 = newPT32 + newPT33 + newPT34;

PTQ12 = newPT35 + newPT36 + newPT37;

PTQ13 = newPT38 + newPT39 + newPT40;

PTQ14 = newPT41 + newPT42 + newPT43;

ONCQ1 = newONC1 + newONC2 + newONC3;

ONCQ2 = newONC4 + newONC5 + newONC6;

ONCQ3 = newONC7 + newONC8 + newONC9;

ONCQ4 = newONC10 + newONC11 + newONC12;

ONCQ5 = newONC13 + newONC14 + newONC15;

ONCQ6 = newONC16 + newONC17 + newONC18;

ONCQ7 = newONC20 + newONC21 + newONC22;

ONCQ8 = newONC23 + newONC24 + newONC25;

ONCQ9 = newONC26 + newONC27 + newONC28;

ONCQ10 = newONC29 + newONC30 + newONC31;

ONCQ11 = newONC32 + newONC33 + newONC34;

ONCQ12 = newONC35 + newONC36 + newONC37;

ONCQ13 = newONC38 + newONC39 + newONC40;

ONCQ14 = newONC41 + newONC42 + newONC43;

DCQ1 = newDC1 + newDC2 + newDC3;

DCQ2 = newDC4 + newDC5 + newDC6;

DCQ3 = newDC7 + newDC8 + newDC9;

DCQ4 = newDC10 + newDC11 + newDC12;

DCQ5 = newDC13 + newDC14 + newDC15;

DCQ6 = newDC16 + newDC17 + newDC18;

DCQ7 = newDC20 + newDC21 + newDC22;

DCQ8 = newDC23 + newDC24 + newDC25;

DCQ9 = newDC26 + newDC27 + newDC28;

DCQ10 = newDC29 + newDC30 + newDC31;

DCQ11 = newDC32 + newDC33 + newDC34;

```

DCQ12 = newDC35 + newDC36 + newDC37;
DCQ13 = newDC38 + newDC39 + newDC40;
DCQ14 = newDC41 + newDC42 + newDC43;
ODCQ1 = newODC1 + newODC2 + newODC3;
ODCQ2 = newODC4 + newODC5 + newODC6;
ODCQ3 = newODC7 + newODC8 + newODC9;
ODCQ4 = newODC10 + newODC11 + newODC12;
ODCQ5 = newODC13 + newODC14 + newODC15;
ODCQ6 = newODC16 + newODC17 + newODC18;
ODCQ7 = newODC20 + newODC21 + newODC22;
ODCQ8 = newODC23 + newODC24 + newODC25;
ODCQ9 = newODC26 + newODC27 + newODC28;
ODCQ10 = newODC29 + newODC30 + newODC31;
ODCQ11 = newODC32 + newODC33 + newODC34;
ODCQ12 = newODC35 + newODC36 + newODC37;
ODCQ13 = newODC38 + newODC39 + newODC40;
ODCQ14 = newODC41 + newODC42 + newODC43;
run;

```

```

/*Find average utilizations, stratified by AF_AMYLOID_RESULT (scan result)*/

```

```

proc sort data=care_codes;
by af_amyloid_result;
run;

```

```

proc means data=care_codes;
var NSCQ3 -NSCQ14 PTQ3 - PTQ14 ONCQ3-ONCQ14 DCQ3 -DCQ14 ODCQ3 - ODCQ14;
class af_amyloid_result;
output out=util_means1;
run;

```

```

data util_means1;
set util_means1;
if _STAT_ = "MEAN" and (af_amyloid_result = 1 or af_amyloid_result = 2) then output util_means1;
keep af_amyloid_result NSCQ3 -NSCQ14 PTQ3 - PTQ14 ONCQ3-ONCQ14 DCQ3 -DCQ14 ODCQ3 -
ODCQ14;
run;

```

```

/*Find average utilizations, stratified by REC_cpait_CrrctIntrtScn_Cat (scan result vs. patient's interpretation of
scan result)*/

```

```

proc sort data=care_codes;
by REC_cpait_CrrctIntrtScn_Cat;
run;

```

```

proc means data=care_codes;
var NSCQ3 -NSCQ14 PTQ3 - PTQ14 ONCQ3-ONCQ14 DCQ3 -DCQ14 ODCQ3 - ODCQ14;
class REC_cpait_CrrctIntrtScn_Cat;
output out=util_means2;
run;

```

```

data util_means2;
set util_means2;
if _STAT_ = "MEAN" and (rec_cpait_crrctintrtscn_cat = 1 or rec_cpait_crrctintrtscn_cat = 2 or
rec_cpait_crrctintrtscn_cat = 3 or rec_cpait_crrctintrtscn_cat = 4) then output util_means2;
keep REC_cpait_CrrctIntrtScn_Cat NSCQ3 -NSCQ14 PTQ3 - PTQ14 ONCQ3-ONCQ14 DCQ3 -DCQ14
ODCQ3 - ODCQ14;
run;

```



```

data quarterly_codes;
set care_codes;
keep bene_id af_amyloid_result REC_cpait_CrrctIntrtScn_Cat NSCQ3 -NSCQ14 PTQ3 - PTQ14 ONCQ3-
ONCQ14 DCQ3 -DCQ14 ODCQ3 - ODCQ14;
run;

data quarterly_codes;
set quarterly_codes (rename=(NSCQ3-NSCQ14 = NSCQ1 - NSCQ12 PTQ3-PTQ14 = PTQ1-PTQ12 ONCQ3-
ONCQ14 = ONCQ1-ONCQ12 DCQ3-DCQ14 = DCQ1 - DCQ12 ODCQ3-ODCQ14 = ODCQ1-ODCQ12));
run;

data sh055289.quarterly_codes;
set quarterly_codes;
ALLQ1 = NSCQ1 + PTQ1 + ONCQ1 + DCQ1 + ODCQ1;
ALLQ2 = NSCQ2 + PTQ2 + ONCQ2 + DCQ2 + ODCQ2;
ALLQ3 = NSCQ3 + PTQ3 + ONCQ3 + DCQ3 + ODCQ3;
ALLQ4 = NSCQ4 + PTQ4 + ONCQ4 + DCQ4 + ODCQ4;
ALLQ5 = NSCQ5 + PTQ5 + ONCQ5 + DCQ5 + ODCQ5;
ALLQ6 = NSCQ6 + PTQ6 + ONCQ6 + DCQ6 + ODCQ6;
ALLQ7 = NSCQ7 + PTQ7 + ONCQ7 + DCQ7 + ODCQ7;
ALLQ8 = NSCQ8 + PTQ8 + ONCQ8 + DCQ8 + ODCQ8;
ALLQ9 = NSCQ9 + PTQ9 + ONCQ9 + DCQ9 + ODCQ9;
ALLQ10 = NSCQ10 + PTQ10 + ONCQ10 + DCQ10 + ODCQ10;
ALLQ11 = NSCQ11 + PTQ11 + ONCQ11 + DCQ11 + ODCQ11;
ALLQ12 = NSCQ12 + PTQ12 + ONCQ12 + DCQ12 + ODCQ12;
run;

proc sql;
create table quarterly_codes as select * from quarterly_codes as a left join sh055289.merged as b
on a.bene_id = b.bene_id;
run;

proc sql;
create table quarterly_codes as select * from quarterly_codes as a left join sh055289.cc_scan_year as b
on a.bene_id = b.bene_id;
run;

/*Determine which beneficiaries died during the study period*/
data death_dates;
set sh055289.survey_and_bsf;
if bene_death_dt = . then delete;
keep bene_id bene_death_dt;
run;

proc sort data=death_dates nodupkey;
by bene_id bene_death_dt;
run;

proc sql;
create table quarterly_codes as select * from quarterly_codes as a left join death_dates as b
on a.bene_id = b.bene_id;
run;

data quarterly_codes;
set quarterly_codes;

```

```

end_post_period = intnx('year', scan_date, 2, 'same');
if bene_death_dt < end_post_period and bene_death_dt ~= . then died = 1;
else died = 0;
run;

```

```

data ffs_patients;
set sh055289.ffs_patients;
run;

```

```

proc sql;
create table out as select * from ffs_patients as a left join quarterly_codes as b
on a.bene_id = b.bene_id;
quit;

```

```

/*Use an anti-join to find all patients who were not joined to CC data*/
proc sql;
create table no_merge as select * from quarterly_codes t1 left join ffs_patients t2
on t1.bene_id = t2.bene_id
where t2.bene_id is NULL;
quit;

```

```

proc sort data=out;
by af_amyloid_result;
run;

```

```

proc sort data=no_merge;
by af_amyloid_result;
run;

```

```

proc freq data=out;
table rec_cpait_race cpait_Gender2 alzh_01 alzh_demen_01 atrial_fib_01
cataract_01 chronickidney_01 copd_01 chf_01 diabetes_01 glaucoma_01
hip_fracture_01 ischemicheart_01 depression_01 osteoporosis_01 ra_oa_01
stroke_tia_01 cancer_breast_01 cancer_colorectal_01 cancer_prostate_01
cancer_lung_01 cancer_endometrial_01 anemia_01 hyperl_01 hypert_01 hyperp_01 hypoth_01 died;
by af_amyloid_result;
output out=demos_enroll;
run;

```

```

proc freq data=no_merge;
table rec_cpait_race cpait_Gender2 alzh_01 alzh_demen_01 atrial_fib_01
cataract_01 chronickidney_01 copd_01 chf_01 diabetes_01 glaucoma_01
hip_fracture_01 ischemicheart_01 depression_01 osteoporosis_01 ra_oa_01
stroke_tia_01 cancer_breast_01 cancer_colorectal_01 cancer_prostate_01
cancer_lung_01 cancer_endometrial_01 anemia_01 hyperl_01 hypert_01 hyperp_01 hypoth_01 died;
by af_amyloid_result;
run;

```

```

proc means data=out;
var REC_cpait_AGE;
by af_amyloid_result;
run;

```

```

proc means data=no_merge;
var REC_cpait_AGE;

```

```
by af_amyloid_result;  
run;
```

```
/*Create new variable that contains the sum of nervous system codes, psychotherapy codes, other nervous  
system codes, diagnostic codes, other diagnostic codes*/
```

```
data out;  
set out;  
ALLQ1 = NSCQ1 + PTQ1 + ONCQ1 + DCQ1 + ODCQ1;  
ALLQ2 = NSCQ2 + PTQ2 + ONCQ2 + DCQ2 + ODCQ2;  
ALLQ3 = NSCQ3 + PTQ3 + ONCQ3 + DCQ3 + ODCQ3;  
ALLQ4 = NSCQ4 + PTQ4 + ONCQ4 + DCQ4 + ODCQ4;  
ALLQ5 = NSCQ5 + PTQ5 + ONCQ5 + DCQ5 + ODCQ5;  
ALLQ6 = NSCQ6 + PTQ6 + ONCQ6 + DCQ6 + ODCQ6;  
ALLQ7 = NSCQ7 + PTQ7 + ONCQ7 + DCQ7 + ODCQ7;  
ALLQ8 = NSCQ8 + PTQ8 + ONCQ8 + DCQ8 + ODCQ8;  
ALLQ9 = NSCQ9 + PTQ9 + ONCQ9 + DCQ9 + ODCQ9;  
ALLQ10 = NSCQ10 + PTQ10 + ONCQ10 + DCQ10 + ODCQ10;  
ALLQ11 = NSCQ11 + PTQ11 + ONCQ11 + DCQ11 + ODCQ11;  
ALLQ12 = NSCQ12 + PTQ12 + ONCQ12 + DCQ12 + ODCQ12;  
ALL_YR1 = ALLQ1 + ALLQ2 + ALLQ3 + ALLQ4;  
ALL_YR2 = ALLQ5 + ALLQ6 + ALLQ7 + ALLQ8;  
ALL_YR3 = ALLQ9 + ALLQ10 + ALLQ11 + ALLQ12;  
run;
```

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

```
data sh055289.regression_data_patients;  
set out;  
run;
```

Syntax for: Creating analytic data set

```
/*Join patient data to patient utilizations outcomes*/  
data regression_patients;  
set sh055289.regression_data_patients;  
run;
```

```
proc sql;  
create table new1 as select * from regression_patients as a left join sh055289.quarterly_utilizations as b  
on a.bene_id = b.bene_id;  
run;
```

```
proc sql;  
create table new2 as select * from new1 as a left join sh055289.quarterly_codes as b  
on a.bene_id = b.bene_id;  
run;
```

```
data SH055289.regression_data_wide;  
set new2;  
run;
```

```
data regression_data_wide;  
set SH055289.regression_data_wide;  
run;
```

```

/*Transpose data for regression (each row is unique combination of bene_id, quarter)*/
proc transpose data= SH055289.regression_data_wide out=er_long prefix=ER;
by bene_id;
var ERQ1-ERQ12;
run;

```

```

proc transpose data= SH055289.regression_data_wide out=ip_long prefix=IP;
by bene_id;
var IPQ1-IPQ12;
run;

```

```

proc transpose data= SH055289.regression_data_wide out=op_long prefix=OP;
by bene_id;
var OPQ1-OPQ12;
run;

```

```

proc transpose data= SH055289.regression_data_wide out=all_long prefix=ALL;
by bene_id;
var ALLQ1-ALLQ12;
run;

```

```

data er_long;
set er_long;
if _NAME_ = "ERQ1" then quarter = 1;
if _NAME_ = "ERQ2" then quarter = 2;
if _NAME_ = "ERQ3" then quarter = 3;
if _NAME_ = "ERQ4" then quarter = 4;
if _NAME_ = "ERQ5" then quarter = 5;
if _NAME_ = "ERQ6" then quarter = 6;
if _NAME_ = "ERQ7" then quarter = 7;
if _NAME_ = "ERQ8" then quarter = 8;
if _NAME_ = "ERQ9" then quarter = 9;
if _NAME_ = "ERQ10" then quarter = 10;
if _NAME_ = "ERQ11" then quarter = 11;
if _NAME_ = "ERQ12" then quarter = 12;
run;

```

```

data ip_long;
set ip_long;
if _NAME_ = "IPQ1" then quarter = 1;
if _NAME_ = "IPQ2" then quarter = 2;
if _NAME_ = "IPQ3" then quarter = 3;
if _NAME_ = "IPQ4" then quarter = 4;
if _NAME_ = "IPQ5" then quarter = 5;
if _NAME_ = "IPQ6" then quarter = 6;
if _NAME_ = "IPQ7" then quarter = 7;
if _NAME_ = "IPQ8" then quarter = 8;
if _NAME_ = "IPQ9" then quarter = 9;
if _NAME_ = "IPQ10" then quarter = 10;
if _NAME_ = "IPQ11" then quarter = 11;
if _NAME_ = "IPQ12" then quarter = 12;
run;

```

```

data op_long;
set op_long;

```

```

if _NAME_ = "OPQ1" then quarter = 1;
if _NAME_ = "OPQ2" then quarter = 2;
if _NAME_ = "OPQ3" then quarter = 3;
if _NAME_ = "OPQ4" then quarter = 4;
if _NAME_ = "OPQ5" then quarter = 5;
if _NAME_ = "OPQ6" then quarter = 6;
if _NAME_ = "OPQ7" then quarter = 7;
if _NAME_ = "OPQ8" then quarter = 8;
if _NAME_ = "OPQ9" then quarter = 9;
if _NAME_ = "OPQ10" then quarter = 10;
if _NAME_ = "OPQ11" then quarter = 11;
if _NAME_ = "OPQ12" then quarter = 12;
run;

```

```

data all_long;
set all_long;
if _NAME_ = "ALLQ1" then quarter = 1;
if _NAME_ = "ALLQ2" then quarter = 2;
if _NAME_ = "ALLQ3" then quarter = 3;
if _NAME_ = "ALLQ4" then quarter = 4;
if _NAME_ = "ALLQ5" then quarter = 5;
if _NAME_ = "ALLQ6" then quarter = 6;
if _NAME_ = "ALLQ7" then quarter = 7;
if _NAME_ = "ALLQ8" then quarter = 8;
if _NAME_ = "ALLQ9" then quarter = 9;
if _NAME_ = "ALLQ10" then quarter = 10;
if _NAME_ = "ALLQ11" then quarter = 11;
if _NAME_ = "ALLQ12" then quarter = 12;
run;

```

```

proc sort data=er_long;
by bene_id quarter;
run;

```

```

proc sort data=ip_long;
by bene_id quarter;
run;

```

```

proc sort data=op_long;
by bene_id quarter;
run;

```

```

proc sort data=all_long;
by bene_id quarter;
run;

```

```

data long_join;
merge er_long (rename=(ER1=ER) drop=_name_)
      ip_long (rename=(IP1=IP) drop=_name_)
      op_long (rename=(OP1=OP) drop=_name_)
      all_long (rename=(ALL1=ALL) drop=_name_);
by bene_id quarter;
drop _name_;

```

```

run;

proc sort data=long_join;
by bene_id quarter;
run;

data reg_data_vars;
set SH055289.regression_data_patients;
keep bene_id rec_cpait_relata rec_cpait_age rec_cpait_race_ethn REC_cpait_EDU_4cat cpait_GenHealth
cpait_RiskDeath cpait_AdvanceDirective impairment_level af_amyloid_result cpait_gender2 rec_cpait_marstat
alz_01 alz_demen_01 atrial_fib_01 cataract_01 chronickidney_01 copd_01 chf_01 diabetes_01
glaucoma_01 ischemicheart_01 depression_01 osteoporosis_01 ra_oa_01 stroke_tia_01 cancer_breast_01
cancer_colorectal_01 cancer_prostate_01 cancer_lung_01 cancer_endometrial_01 cancer_all_01 anemia_01
hyperl_01 hyperp_01 hypert_01 hypoth_01;
if cancer_breast_01 = 1 or cancer_colorectal_01 = 1 or cancer_prostate_01 = 1 or cancer_lung_01 = 1 or
cancer_endometrial_01 = 1 then cancer_all_01 = 1;
else cancer_all_01 = 0;
run;

proc sql;
create table SH055289.quarterly_regression_data as select * from long_join as a left join reg_data_vars as b
on a.bene_id = b.bene_id;
quit;

proc sort data= regression_data_wide;
by af_amyloid_result;
run;

/*Calculate quarterly means of utilization variables*/
proc means data=regression_data_wide;
var ERQ1 -ERQ12 IPQ1 - IPQ12 OPQ1-OPQ12 ALLQ1-ALLQ12;
class af_amyloid_result;
output out=util_means2;
run;

data util_means2;
set util_means2;
if _stat_ ~= "MEAN" then delete;
run;

/*Create yearly utilization columns from quarterly utilization columns*/
data new2;
set regression_data_wide;
ALL_IP1 = IPQ1 + IPQ2 + IPQ3 + IPQ4;
ALL_IP2 = IPQ5 + IPQ6 + IPQ7 + IPQ8;
ALL_IP3 = IPQ9 + IPQ10 + IPQ11 + IPQ12;
ALL_OP1 = OPQ1 + OPQ2 + OPQ3 + OPQ4;
ALL_OP2 = OPQ5 + OPQ6 + OPQ7 + OPQ8;
ALL_OP3 = OPQ9 + OPQ10 + OPQ11 + OPQ12;
ALL_ER1 = ERQ1 + ERQ2 + ERQ3 +ERQ4;
ALL_ER2 = ERQ5 + ERQ6 + ERQ7 + ERQ8;
ALL_ER3 = ERQ9 + ERQ10 + ERQ11 + ERQ12;
ALL_TESTS1 = ALLQ1 + ALLQ2 + ALLQ3 + ALLQ4;
ALL_TESTS2 = ALLQ5 + ALLQ6 + ALLQ7 + ALLQ8;
ALL_TESTS3 = ALLQ9 + ALLQ10 + ALLQ11 + ALLQ12;
run;

```

```

proc sort data=new2;
by bene_id;
run;
/*Transpose yearly utilization data for joining with patient data*/
proc transpose data=new2 out=ip_data;
by bene_id;
var all_ip1-all_ip3;
run;

```

```

data ip_data;
set ip_data;
rename col1 = inpatient_claims;
if _NAME_ = "ALL_IP1" then pre_post = "ALL_YR1";
if _NAME_ = "ALL_IP2" then pre_post = "ALL_YR2";
if _NAME_ = "ALL_IP3" then pre_post = "ALL_YR3";
run;

```

```

proc transpose data=new2 out=op_data;
by bene_id;
var all_op1-all_op3;
run;

```

```

data op_data;
set op_data;
rename col1 = outpatient_claims;
if _NAME_ = "ALL_OP1" then pre_post = "ALL_YR1";
if _NAME_ = "ALL_OP2" then pre_post = "ALL_YR2";
if _NAME_ = "ALL_OP3" then pre_post = "ALL_YR3";
run;

```

```

proc transpose data=new2 out=er_data;
by bene_id;
var all_er1-all_er3;
run;

```

```

data er_data;
set er_data;
rename col1 = emergency_claims;
if _NAME_ = "ALL_ER1" then pre_post = "ALL_YR1";
if _NAME_ = "ALL_ER2" then pre_post = "ALL_YR2";
if _NAME_ = "ALL_ER3" then pre_post = "ALL_YR3";
run;

```

```

proc transpose data=new2 out=all_tests;
by bene_id;
var all_tests1-all_tests3;
run;

```

```

data all_tests;
set all_tests;
rename col1 = test_claims;
if _NAME_ = "ALL_TESTS1" then pre_post = "ALL_YR1";
if _NAME_ = "ALL_TESTS2" then pre_post = "ALL_YR2";
if _NAME_ = "ALL_TESTS3" then pre_post = "ALL_YR3";

```

```

run;

/*Join each of the utilization columns to the patient data*/
data regression_data_alz_tests;
set sh055289.regression_data_alz_tests;
run;

proc sql;
create table claim_set as select * from er_data as a left join ip_data as b
on a.bene_id = b.bene_id and a.pre_post = b.pre_post;
quit;

proc sql;
create table claim_set as select * from claim_set as a left join op_data as b
on a.bene_id = b.bene_id and a.pre_post = b.pre_post;
quit;

proc sql;
create table claim_set as select * from claim_set as a left join all_tests as b
on a.bene_id = b.bene_id and a.pre_post = b.pre_post;
quit;

data claim_set;
set claim_set;
keep bene_id emergency_claims inpatient_claims outpatient_claims test_claims pre_post;
run;

proc sql;
create table SH055289.new_regression_data as select * from claim_set as a left join
SH055289.regression_data_patients as b
on a.bene_id = b.bene_id;
quit;

```

Syntax for: Creating Table 1 (Stata)

```

//Categorical variables
tab af_amyloid_result2
tab cpait_gender2 af_amyloid_result2, chi2
tab race af_amyloid_result2, chi2
tab education af_amyloid_result2, chi2
tab rec_cpait_marstat af_amyloid_result2, chi2
tab q_rd af_amyloid_result2, chi2
tab bd_htn af_amyloid_result2, chi2
tab n_afib af_amyloid_result2, chi2
tab bd_vhd af_amyloid_result2, chi2
tab q_copd af_amyloid_result2, chi2
tab bd_dm af_amyloid_result2, chi2
tab q_depres af_amyloid_result2, chi2
tab bd_stk af_amyloid_result2, chi2
tab any_chf af_amyloid_result2, chi2
tab q_psych af_amyloid_result2, chi2
tab q_mi af_amyloid_result2, chi2
tab any_cancer af_amyloid_result2, chi2

```



```

tab advdir af_amyloid_result2, chi2
tab cgspse af_amyloid_result2, chi2
tab genhlth af_amyloid_result2, chi2
tab rskdth af_amyloid_result2, chi2
tab impair af_amyloid_result2, chi2

```

```

//Continuous variables
ttest age, by(af_amyloid_result2)
ttest charlson, by(af_amyloid_result2)
ttest time_diff, by(af_amyloid_result2)

```

Syntax for: Regression analysis (Stata)

```
//Main
```

```
import delimited "U:\DUA_055289\Shared\PTS\NEW_REGRESSION_DATA2.csv", clear
```

```
log using revised_final_models.log
```

```
encode pre_post, generate(new_pre_post)
```

```

gen cgspse = .
replace cgspse = 1 if rec_cpait_relat == "1"
replace cgspse = 0 if rec_cpait_relat != "1"

```

```
gen age = real(rec_cpait_age)
```

```

encode rec_cpait_race_ethn, generate(race)
encode rec_cpait_edu_4cat, generate(education)

```

```

gen genhlth = .
replace genhlth = 1 if cpait_genhealth == "4" | cpait_genhealth == "5"
replace genhlth = 0 if cpait_genhealth == "1" | cpait_genhealth == "2" | cpait_genhealth == "3"

```

```

replace cpait_riskdeath = "0" if cpait_riskdeath == "R" | cpait_riskdeath == "D"
gen int_riskdeath = real(cpait_riskdeath)
gen rskdth = .
replace rskdth = 0 if int_riskdeath >= 0 & int_riskdeath <= 49
replace rskdth = 1 if int_riskdeath >= 50

```

```

gen advdir = .
replace advdir = 1 if cpait_advancedirective == "1"
replace advdir = 0 if cpait_advancedirective == "0"

```

```

gen cancer_all_01 = .
replace cancer_all_01 = 1 if (cancer_breast == 1 | cancer_colorectal_01 == 1 | cancer_prostate_01 == 1 |
cancer_lung_01 == 1 | cancer_endometrial_01 == 1)
replace cancer_all_01 = 0 if cancer_all_01 == .

```

```

gen af_amyloid_result2 = .
replace af_amyloid_result2 = 0 if af_amyloid_result == 2
replace af_amyloid_result2 = 1 if af_amyloid_result == 1

```

```

gen impair = .
replace impair = 1 if impairment_level == 2

```

replace impair = 0 if impairment_level == 1

drop if rec_cpait_age == "R"

drop if rec_cpait_age == "D"

drop if advdir == .

drop if rskdth == .

drop if genhlth == .

//Outpatient

nbreg outpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2 age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)

margins i.new_pre_post##i.af_amyloid_result2

margins af_amyloid_result2, dydx(new_pre_post)

//Inpatient

nbreg inpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2 age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)

margins i.new_pre_post##i.af_amyloid_result2

margins af_amyloid_result2, dydx(new_pre_post)

//Emergency

nbreg emergency_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2 age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)

margins i.new_pre_post##i.af_amyloid_result2

margins af_amyloid_result2, dydx(new_pre_post)

//Tests

nbreg test_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2 age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)

margins i.new_pre_post##i.af_amyloid_result2

margins af_amyloid_result2, dydx(new_pre_post)

//RISK DEATH

//Outpatient

nbreg outpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.rskdth
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth impair time_diff, irr
vce(cluster bene_id)

margins af_amyloid_result2##rskdth##new_pre_post

margins rskdth##af_amyloid_result2, dydx(new_pre_post)

//Inpatient

```

nbreg inpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.rskdth age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth impair time_diff, irr vce(cluster
bene_id)
margins af_amyloid_result2##rskdth##new_pre_post
margins rskdth##af_amyloid_result2, dydx(new_pre_post)

```

//Emergency

```

nbreg emergency_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.rskdth
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth impair time_diff, irr
vce(cluster bene_id)
margins af_amyloid_result2##rskdth##new_pre_post
margins rskdth##af_amyloid_result2, dydx(new_pre_post)

```

//Tests

```

nbreg test_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.rskdth age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse genhlth impair time_diff, irr vce(cluster
bene_id)
margins af_amyloid_result2##rskdth##new_pre_post
margins rskdth##af_amyloid_result2, dydx(new_pre_post)

```

//GENERAL HEALTH

//Outpatient

```

nbreg outpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.genhlth
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse rskdth impair time_diff, irr
vce(cluster bene_id)

```

```

margins i.new_pre_post##i.af_amyloid_result2##i.genhlth
margins genhlth##af_amyloid_result2, dydx(new_pre_post)

```

//Inpatient

```

nbreg inpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.genhlth
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse rskdth impair time_diff, irr
vce(cluster bene_id)

```

```

margins i.new_pre_post##i.af_amyloid_result2##i.genhlth
margins genhlth##af_amyloid_result2, dydx(new_pre_post)

```

//Emergency

```

nbreg emergency_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.genhlth
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse rskdth impair time_diff, irr
vce(cluster bene_id)

```

```

margins i.new_pre_post##i.af_amyloid_result2##i.genhlth
margins genhlth##af_amyloid_result2, dydx(new_pre_post)

```

//Test

```

nbreg test_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.genhlth age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson advdir cgspse rskdth impair time_diff, irr vce(cluster
bene_id)

```

```
margins i.new_pre_post##i.af_amyloid_result2##i.genhlth
margins genhlth##af_amyloid_result2, dydx(new_pre_post)
```

//ADVANCE DIRECTIVE

//Outpatient

```
nbreg outpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.advdir
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)
```

```
margins new_pre_post##af_amyloid_result2##advdir
margins advdir##af_amyloid_result2, dydx(new_pre_post)
```

//Inpatient

```
nbreg inpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.advdir age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth impair time_diff, irr vce(cluster
bene_id)
```

```
margins new_pre_post##af_amyloid_result2##advdir
margins advdir##af_amyloid_result2, dydx(new_pre_post)
```

//Emergency

```
nbreg emergency_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.advdir
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth impair time_diff, irr
vce(cluster bene_id)
```

```
margins new_pre_post##af_amyloid_result2##advdir
margins advdir##af_amyloid_result2, dydx(new_pre_post)
```

//Tests

```
nbreg test_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.advdir age
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk
any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth impair time_diff, irr vce(cluster
bene_id)
```

```
margins new_pre_post##af_amyloid_result2##advdir
margins advdir##af_amyloid_result2, dydx(new_pre_post)
```

//IMPAIRMENT

//Outpatient

```
nbreg outpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.impair
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth advdir time_diff, irr
vce(cluster bene_id)
```

```
margins new_pre_post##af_amyloid_result2##impair
margins impair##af_amyloid_result2, dydx(new_pre_post)
```

//Inpatient

```
nbreg inpatient_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.impair age  
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk  
any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth advdir time_diff, irr vce(cluster  
bene_id)
```

```
margins new_pre_post##af_amyloid_result2##impair  
margins impair##af_amyloid_result2, dydx(new_pre_post)
```

//Emergency

```
nbreg emergency_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.impair  
age cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres  
bd_stk any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth advdir time_diff, irr  
vce(cluster bene_id)
```

```
margins new_pre_post##af_amyloid_result2##impair  
margins impair##af_amyloid_result2, dydx(new_pre_post)
```

//Tests

```
nbreg test_claims i.new_pre_post i.af_amyloid_result2 i.new_pre_post##af_amyloid_result2##i.impair age  
cpait_gender2 race education rec_cpait_marstat q_rd bd_htn n_afib bd_vhd q_copd bd_dm q_depres bd_stk  
any_chf q_psych q_mi n_epilep any_cancer charlson cgspse genhlth rskdth advdir time_diff, irr vce(cluster  
bene_id)
```

```
margins new_pre_post##af_amyloid_result2##impair  
margins impair##af_amyloid_result2, dydx(new_pre_post)
```

log close