NHATS Technical Paper #13

NATIONAL HEALTH AND AGING TRENDS STUDY (NHATS)

Addendum to Classification of Persons by Dementia Status in the National Health and Aging Trends Study for Rounds 2-4

October 22, 2015

Suggested Citation: Kasper, Judith D., Freedman, Vicki A., Spillman, Brenda C., and Skehan, Maureen E. 2015. Addendum to Classification of Persons by Dementia Status in the National Health and Aging Trends Study for Rounds 2-4. Technical Paper #13. Baltimore: Johns Hopkins University School of Public Health. Available at www.NHATS.org. This technical paper was prepared with funding from the National Institute on Aging (U01AG032947).

This technical paper updates classification of persons by dementia status in the National Health and Aging Trends Study for rounds 2 through 4. Kasper et al 2013 (Technical Paper #5) describes the types of information that the National Health and Aging Trends Study (NHATS) provides to identify persons with cognitive impairment and our approach to classifying persons as having dementia. Stata and SAS programming statements to create a dementia classification variable in Round 1 are available at the NHATS website (www.nhats.org). This paper describes the information collected in follow-up rounds and programming statements for creating a dementia classification variable in Rounds 2 through 4 of NHATS.

NHATS items for classification of persons with dementia

Three types of information are collected in all rounds of NHATS and can be used to identify persons who are cognitively impaired:

- A report by the sample person or proxy respondent that a doctor told the sample person that he/she had dementia or Alzheimer's disease.

- A score that indicates probable dementia on the AD8 Dementia Screening Interview, which is administered to proxy respondents who are answering the NHATS interview for the sample person. This 8-item instrument assesses memory, temporal orientation, judgment and function (Galvin et al. 2005, 2006).

- Cognitive tests that evaluate the sample person's memory (immediate and delayed 10-word recall), orientation (date, month, year, and day of the week; naming the President and Vice President), and executive function (clock drawing test). For more details on items and administration see Kasper and Freedman (2015); NHATS Data Collection Procedures (2011).

The information available varies by type of respondent: for self-respondents, report of a diagnosis and cognitive test items; when proxy respondents are used report of a diagnosis, responses to the AD8, and cognitive test results if the proxy said the sample person could be asked these questions.

Details regarding the criteria for dementia classification are in Kasper et al. 2013 (Technical Paper #5) as are results from a sensitivity and specificity analysis against a clinically evaluated sample (Aging, Demographics, and Memory Study (ADAMS) Wave E conducted in 2010; Langa et al. 2005) and comparisons with other population-based estimates of dementia. The criteria and cutpoints developed in Round 1 remain unchanged in Rounds 2 through 4.

Changes to Programming Statements for classification of persons with dementia in Rounds 2 to 4

Because NHATS is a longitudinal study that interviews the same individuals every year, how the information used in the dementia classification variable is collected across rounds changes somewhat across rounds. These changes affect programming statements in the following ways:

- Once a sample person or proxy respondent reports that a doctor told the sample person that he/she had dementia or Alzheimer's disease, this question is not re-asked. Instead it is coded as "previously reported" in all rounds following the round of the initial report.

- The AD8 items, which ask about a change in thinking or a memory problem, are asked of proxy respondents at each round. However these items are not asked if *in a prior round* a proxy respondent says that the sample person has dementia or Alzheimer's disease in response to any of the AD8 items. A derived variable (cp5dad8dem) has been created in Rounds 2 through 4, which indicates such a response.

Programming statements for Rounds 2 through are attached. These take into account the interview changes above.

References

Galvin JE, Roe CM, Powlishta KK, Coats MA, Muich SJ, et al. (2005) The AD8: A brief informant interview to detect dementia. Neurology, 65(4): 559-564.

Galvin JE, Roe CM, Xiong C, Morris JC. (2006). Validity and reliability of the AD8 informant interview in dementia. Neurology, 67(11): 1942-1948.

Kasper JD, Freedman VA. 2015. National Health and Aging Trends Study User Guide: Rounds 1, 2, 3 & 4 Final Release. Baltimore: Johns Hopkins University School of Public Health. Available

Kasper JD, Freedman VA, Spillman BC. 2013. Classification of Persons by Dementia Status in the National Health and Aging Trends Study. Technical Paper #5. Baltimore: Johns Hopkins University School of Public Health. Available at www.NHATS.org. at www.nhats.org.

Langa KM, Plassman BL, Wallace RB, Herzog AR, Heeringa SG et al. (2005) The Aging, Demographics, and Memory Study: Study Design and Methods. Neuroepidemiology, 25:181-191.

SAS Programming Statements for Rounds 2
Rounds 3 and 4 can be programmed by replacing all round 2 variables with the appropriate round.

```
** NOTE: The input file to run this code is the NHATS_Round_2_SP_File**
** FORMATS FOR CONSTRUCTED VARIABLES **;
proc format;
value r2demclas
 1="1:Probable dementia"
 2="2:Possible dementia"
 3="3:No dementia"
 -1="-1:Deceased or nursing home resident"
 -9="-9:Missing"
 ;
value r2ad8dem
 1="1:Meets dementia criteria"
 2="2:Does not meet dementia criteria"

value r2clockf
0-1="0-1:Impaired"
2-5="2-5:Not impaired"
;
value r2wordrecf
0-3="0-3: Impaired "
4-20="4-20: Not impaired "
;
value r2dateprf
0-3="0-3: Impaired "
4-8="4-8: Not impaired "
;
** DATE STEP CODE FOR CREATING DEMENTIA CLASSIFICATION VARIABLE **;
length r2demclas
 r2ad8_dem r2ad8_1-r2ad8_8 r2ad8miss_1-r2ad8miss_8 r2ad8_score r2ad8_miss
 r2date_mon r2date_day r2date_yr r2date_dow r2date_sum r2date_sumr
 r2preslast r2presfirst r2vplast r2vpfirst r2presvp r2presvpr r2date_prvp
 r2clock_scorer r2irecall r2drecall r2wordrecall0_20
 r2clock65 r2word65 r2datena65 r2domain65
 3;
label r2demclas="R2 NHATS Dementia Classification 65+";
** 1) SET MISSING (RESIDENTIAL CARE FQ ONLY) AND N.A. (NURSING HOME RESIDENTS, DECEASED) **;
if r2dresid=7 then r2demclas=-9;
if r2dresid in (6,8) then r2demclas=-1;
** 2) CODE PROBABLE IF DEMENTIA DIAGNOSIS REPORTED BY SELF OR PROXY **;
if hc2disescn9 in (1,7) and is2resptype in (1,2) then r2demclas=1;
** 3a) CODE AD8_SCORE **;
array r2think {*} cp2chgthink1-cp2chgthink8; ** QUESTIONNAIRE ITEMS **;
array r2ad8item {*} r2ad8_1-r2ad8_8;
array r2ad8miss {*} r2ad8miss_1-r2ad8miss_8;
```

```
 r2ad8_score =-1;
 r2ad8_miss =-1;

 do i=1 to dim(r2ad8item);
 ** INITIALIZE COUNTS TO NOT APPLICABLE**;
 r2ad8item{i}=-1;
 r2ad8miss{i}=-1;
 ** ASSIGN VALUES TO AD8 ITEMS IF PROXY AND DEMENTIA CLASS NOT ALREADY ASSIGNED BY
REPORTED DIAGNOSIS **;
 if is2resptype=2 and r2demclas=. then do;
 r2ad8item{i}=.;
 if r2think{i} in (1,3) then r2ad8item{i}=1; ** PROXY REPORTS A CHANGE OR ALZ/DEMENTIA **;
 else if r2think{i}=2 then r2ad8item{i}=0; ** PROXY REPORTS NO CHANGE **;
 r2ad8_score=sum(of r2ad8item{*}); ** COUNT AD8 ITEMS **;

 if r2ad8item{i} in (0,1) then r2ad8miss{i}=0;
 else if r2ad8item{i}=. then r2ad8miss{i}=1;
 r2ad8_miss=sum(of r2ad8miss{*}); ** COUNT MISSING AD8 ITEMS **;
 end;
 end;
** 3b) CODE AD8 DEMENTIA CLASS **;
if cp2dad8dem=1 and is2resptype=2 and r2demclas=. then r2ad8_score=8;
if r2ad8_score>=2 then r2ad8_dem=1 ; ** IF SCORE >=2 THEN MEETS AD8 CRITERION **;
if r2ad8_score in (0,1) or (r2ad8_miss=8 and r2ad8_dem=.) then r2ad8_dem=2; ** IF SCORE IS 0 OR 1
OR ALL ITEMS MISSING
THEN DOES NOT MEET AD8 CRITERION **;
** 4) UPDATE DEMENTIA CLASSIFICATION VARIABLE WITH AD8 CLASS **;
if r2demclas=. then do;
 if r2ad8_dem=1 then r2demclas=1; ** PROBABLE BASED ON AD8 SCORE **;
 if r2ad8_dem=2 and cg2speaktosp=2 then r2demclas=3; ** NO DIAGNOSIS, DOES NOT MEET AD8
CRITERION, AND PROXY SAYS CANNOT ASK SP COGNITIVE ITEMS **;
end;
** 5) CODE DATE ITEMS AND COUNT **;
array cg2date {*} cg2todaydat1-cg2todaydat4;
array r2date_item {*} r2date_mon r2date_day r2date_yr r2date_dow;

do i=1 to dim(r2date_item);
 if cg2date{i} > 0 then r2date_item{i}=cg2date{i}; *** CODE ONLY YES/NO RESPONSES: MISSING/N.A.
CODES -1,-9 LEFT MISSING **;
 if cg2date{i} in (-7,2) then r2date_item{i}=0; ** 2:NO/DK OR -7:REFUSED RECODED TO 0:NO/DK/RF
**;
 r2date_sum=sum(of r2date_item{*}); ** COUNT CORRECT DATE ITEMS
**;
end;
if r2date_sum=. then do;
 if cg2speaktosp=2 then r2date_sum=-2; ** PROXY SAYS CAN'T SPEAK TO SP
**;
 else if cg2speaktosp=1 and max(of cg2date{*})=-1 then r2date_sum=-3; ** PROXY SAYS CAN SPEAK TO
```

```
SP BUT SP UNABLE TO ANSWER **;
end;
r2date_sumr=r2date_sum;
if r2date_sum=-2 then r2date_sumr=.; ** MISSING IF PROXY SAYS CAN'T SPEAK TO SP
**;
else if r2date_sum=-3 then r2date_sumr=0; ** 0 IF SP UNABLE TO ANSWER **;
** 6) PRESIDENT AND VICE PRESIDENT NAME ITEMS AND COUNT **;
array cg2pres {*} cg2presidna1 cg2presidna3 cg2vpname1 cg2vpname3;
array r2pres_item {*} r2preslast r2presfirst r2vplast r2vpfirst;
do i=1 to dim(r2pres_item);
 if cg2pres{i} > 0 then r2pres_item{i}=cg2pres{i}; ** CODE ONLY YES/NO RESPONSES: MISSING/N.A.
CODES
-1,-9 LEFT MISSING **;
 if cg2pres{i} in (-7,2) then r2pres_item{i}=0; ** 2:NO/DK OR -7:REFUSED RECODED TO 0:NO/DK/RF
**;
 r2presvp=sum(of r2pres_item{*}); ** COUNT CORRECT PRESIDENT/VEEP NAME ITEMS
**;
end;
if r2presvp=. then do;
 if cg2speaktosp=2 then r2presvp=-2; ** PROXY SAYS CAN'T SPEAK TO SP
**;
 else if cg2speaktosp=1 and max(of cg2pres{*})=-1 then r2presvp=-3; ** PROXY SAYS CAN SPEAK TO SP
BUT SP UNABLE TO ANSWER **;
end;
r2presvpr=r2presvp;
if r2presvp=-2 then r2presvpr=.; ** MISSING IF PROXY SAYS CAN'T SPEAK TO SP **;
else if r2presvp=-3 then r2presvpr=0; ** 0 IF SP UNABLE TO ANSWER **;
** 7) ORIENTATION DOMAIN: SUM OF DATE RECALL AND PRESIDENT NAMING **;
r2date_prvp=sum(r2date_sumr,r2presvpr);
** 8) EXECUTIVE FUNCTION DOMAIN: CLOCK DRAWING SCORE **;
r2clock_scorer=cg2dclkdraw;
if cg2dclkdraw in (-2,-9) then r2clock_scorer=.;
if cg2dclkdraw in (-3,-4,-7) then r2clock_scorer=0;
** IMPUTE MEAN SCORE TO PERSONS MISSING A CLOCK **;
if cg2dclkdraw=-9 and cg2speaktosp=1 then r2clock_scorer=2; ** IF PROXY SAID CAN ASK SP **;
if cg2dclkdraw=-9 and cg2speaktosp=-1 then r2clock_scorer=3; ** IF SELF RESPONDENT **;
** 9) MEMORY DOMAIN: IMMEDIATE AND DELAYED WORD RECALL **;
array cg2recall {*} cg2dwrdimmrc cg2dwrddlyrc;
array r2word_recall {*} r2irecall r2drecall;
do i=1 to dim(r2word_recall);
 r2word_recall{i}=cg2recall{i};
 if cg2recall{i} in (-2,-1) then r2word_recall{i}=.;
 if cg2recall{i} in (-7,-3) then r2word_recall{i}=0;
 r2wordrecall0_20=sum(of r2word_recall{*});
end;
** 10) CREATE COGNITIVE DOMAINS FOR ALL ELIGIBLE **;
** I.E. PROXY BUT PROXY SAYS CAN ASK SP, NOT FQ ONLY, NOT NH **;
if 1 < r2clock_scorer <=5 then r2clock65=0;
```

```
if 0 <= r2clock_scorer <=1 then r2clock65=1;
if 3 < r2wordrecall0_20 <=20 then r2word65=0;
if 0 <= r2wordrecall0_20 <=3 then r2word65=1;
if 3 < r2date_prvp <= 8 then r2datena65=0;
if 0 <= r2date_prvp <= 3 then r2datena65=1;
** 10) CREATE COGNITIVE DOMAIN SCORE **;
array r2domains {*} r2clock65 r2word65 r2datena65;
do i=1 to dim(r2domains);
 r2domain65=sum(of r2domains{*});
end;
** 11) UPDATE COGNITIVE CLASSIFICATION **;

if r2demclas=. and cg2speaktosp in (-1,1) then do;
if 2 <= r2domain65 <=3 then r2demclas=1; ** PROBABLE DEMENTIA **;
if r2domain65 =1 then r2demclas=2; ** POSSIBLE DEMENTIA **;
if r2domain65 =0 then r2demclas=3; ** NO DEMENTIA **;

end;
```

Stata programming statements for Round 2
Rounds 3 and 4 can be programmed by replacing all round 2 variables with the appropriate round

```
** NOTE: The input file to run this code is the NHATS_Round_2_SP_File**
*SET MISSING (RESIDENTIAL CARE FQ ONLY) AND N.A. (NURSING HOME RESIDENTS, DECEASED)*
gen r2demclas=-9 if r2dresid==7
replace r2demclas=-1 if r2dresid==6 |r2dresid==8
*CODE PROBABLE IF DEMENTIA DIAGNOSIS REPORTED BY SELF OR PROXY*
replace r2demclas=1 if (hc2disescn9==1 | hc2disescn9==7) & (is2resptype==1 | is2resptype==2)
tab r2demclas
*CODE AD8_SCORE*
*INITIALIZE COUNTS TO NOT APPLICABLE*
*ASSIGN VALUES TO AD8 ITEMS IF PROXY AND DEMENTIA CLASS NOT ALREADY ASSIGNED BY REPORTED
DIAGNOSIS
foreach num of numlist 1/8 {
    *INITIALIZE COUNTS TO NOT APPLICABLE*
          gen r2ad8_`num'=-1
          replace r2ad8_`num'=. if is2resptype==2 & r2demclas==.
          *PROXY REPORTS A CHANGE OR ALZ/DEMENTIA*
    replace r2ad8_`num'=1 if is2resptype==2 & r2demclas==. & (cp2chgthink`num'==1 |
cp2chgthink`num'==3)
          *PROXY REPORTS NO CHANGE*
    replace r2ad8_`num'=0 if is2resptype==2 & r2demclas==. & (cp2chgthink`num'==2) & r2ad8_`num'==.
}

foreach num of numlist 1/8 {
          *INITIALIZE COUNTS TO NOT APPLICABLE*
          gen r2ad8miss_`num'=-1
          replace r2ad8miss_`num'=0 if is2resptype==2 & r2demclas==. & (r2ad8_`num'==0 |
r2ad8_`num'==1)
   replace r2ad8miss_`num'=1 if is2resptype==2 & r2demclas==. & r2ad8_`num'==.
          replace r2ad8_`num'=0 if is2resptype==2 & r2demclas==. & r2ad8_`num'==.
}

*COUNT AD8 ITEMS*
gen r2ad8_score=-1
replace r2ad8_score=(r2ad8_1+r2ad8_2+r2ad8_3+r2ad8_4+r2ad8_5+r2ad8_6+r2ad8_7+r2ad8_8) if
is2resptype==2 & r2demclas==.
*SET PREVIOUS ROUND DEMENTIA DIAGNOSIS BASED ON AD8 TO AD8_SCORE=8*
replace r2ad8_score=8 if cp2dad8dem==1 & is2resptype==2 & r2demclas==.
*COUNT MISSING AD8 ITEMS*
gen r2ad8_miss= -1
replace
r2ad8_miss=(r2ad8miss_1+r2ad8miss_2+r2ad8miss_3+r2ad8miss_4+r2ad8miss_5+r2ad8miss_6+r2ad8
miss_7+r2ad8miss_8) if is2resptype==2 & r2demclas==.

*CODE AD8 DEMENTIA CLASS*
*IF SCORE>=2 THEN MEETS AD8 CRITERIA*
```

```
gen r2ad8_dem=1 if r2ad8_score>=2
* IF SCORE IS 0 OR 1 OR ALL ITEMS MISSING THEN DOES NOT MEET AD8 CRITERION*
replace r2ad8_dem=2 if (r2ad8_score==0 | r2ad8_score==1 | r2ad8_miss==8) & r2ad8_dem==.

*UPDATE DEMENTIA CLASSIFICATION VARIABLE WITH AD8 CLASS*
*PROBABLE DEMENTIA BASED ON AD8 SCORE*
replace r2demclas=1 if r2ad8_dem==1 & r2demclas==.
*NO DIAGNOSIS, DOES NOT MEET AD8 CRITERION, AND PROXY SAYS CANNOT ASK SP COGNITIVE
ITEMS*
replace r2demclas=3 if r2ad8_dem==2 & cg2speaktosp==2 & r2demclas==.
tab r2demclas

*CODE DATE ITEMS AND COUNT*
foreach num of numlist 1/4 {
        *CODE ONLY YES/NO RESPONSES: MISSING/NA CODES -1, -9 LEFT MISSING*
        gen r2date_item`num'=cg2todaydat`num' if cg2todaydat`num'>0
        *2: NO/DK OR -7: REFUSED RECODED TO : NO/DK/RF*
        replace r2date_item`num'=0 if cg2todaydat`num'==2 | cg2todaydat`num'==-7
}

*COUNT CORRECT DATE ITEMS*
gen r2date_sum=r2date_item1 + r2date_item2 + r2date_item3 + r2date_item4
*PROXY SAYS CAN'T SPEAK TO SP*
replace r2date_sum=-2 if r2date_sum==. & cg2speaktosp==2
*PROXY SAYS CAN SPEAK TO SP BUT SP UNABLE TO ANSWER*
replace r2date_sum=-3 if (r2date_item1==. | r2date_item2==. | r2date_item3==. | r2date_item4==.) &
cg2speaktosp==1

gen r2date_sumr=r2date_sum
*MISSING IF PROXY SAYS CAN'T SPEAK TO SP*
replace r2date_sumr=. if r2date_sum==-2
*0 IF SP UNABLE TO ANSWER*
replace r2date_sumr=0 if r2date_sum==-3

*PRESIDENT AND VICE PRESIDENT NAME ITEMS AND COUNT*
** CODE ONLY YES/NO RESPONSES: MISSING/N.A. CODES -1,-9 LEFT MISSING *
*2:NO/DK OR -7:REFUSED RECODED TO 0:NO/DK/RF*
gen r2preslast=cg2presidna1 if cg2presidna1>0
replace r2preslast=0 if cg2presidna1==-7 | cg2presidna1==2

gen r2presfirst=cg2presidna3 if cg2presidna3>0
replace r2presfirst=0 if cg2presidna3==-7 | cg2presidna3==2

gen r2vplast=cg2vpname1 if cg2vpname1>0
replace r2vplast=0 if cg2vpname1==-7 | cg2vpname1==2

gen r2vpfirst=cg2vpname3 if cg2vpname3>0
replace r2vpfirst=0 if cg2vpname3==-7 | cg2vpname3==2
```

```
*COUNT CORRECT PRESIDENT/VP NAME ITEMS*
gen r2presvp= r2preslast+r2presfirst+r2vplast+r2vpfirst
** PROXY SAYS CAN'T SPEAK TO SP *
replace r2presvp=-2 if r2presvp==. & cg2speaktosp==2
** PROXY SAYS CAN SPEAK TO SP BUT SP UNABLE TO ANSWER *
replace r2presvp=-3 if r2presvp==. & cg2speaktosp==1 & (r2preslast==. | r2presfirst==. | r2vplast==. |
r2vpfirst==.)
gen r2presvpr=r2presvp
*MISSING IF PROXY SAYS CAN'T SPEAK TO SP*
replace r2presvpr=. if r2presvp==-2
*0 IF SP UNABLE TO ANSWER*
replace r2presvpr=0 if r2presvp==-3

*ORIENTATION DOMAIN: SUM OF DATE RECALL AND PRESIDENT/VP NAMING*
gen r2date_prvp=r2date_sumr + r2presvpr

*EXECUTIVE FUNCTION DOMAIN: CLOCK DRAWING SCORE*
gen r2clock_scorer=cg2dclkdraw
replace r2clock_scorer=. if cg2dclkdraw==-2 | cg2dclkdraw==-9
replace r2clock_scorer=0 if cg2dclkdraw==-3 | cg2dclkdraw==-4 | cg2dclkdraw==-7
*IMPUTE MEAN SCORE TO PERSONS MISSING A CLOCK*
*IF PROXY SAID CAN ASK SP*
replace r2clock_scorer=2 if cg2dclkdraw==-9 & cg2speaktosp==1
*IF SELF-RESPONDENT*
replace r2clock_scorer=3 if cg2dclkdraw==-9 & cg2speaktosp==-1

*MEMORY DOMAIN: IMMEDIATE AND DELAYED WORD RECALL*
gen r2irecall=cg2dwrdimmrc
replace r2irecall=. if cg2dwrdimmrc==-2 | cg2dwrdimmrc==-1
replace r2irecall=0 if cg2dwrdimmrc==-7 | cg2dwrdimmrc==-3

gen r2drecall=cg2dwrddlyrc
replace r2drecall=. if cg2dwrddlyrc==-2 | cg2dwrddlyrc==-1
replace r2drecall=0 if cg2dwrddlyrc==-7 | cg2dwrddlyrc==-3

gen r2wordrecall0_20=r2irecall+r2drecall

*CREATE COGNITIVE DOMAINS FOR ALL ELIGIBLE*
gen r2clock65=0 if r2clock_scorer>1 & r2clock_scorer<=5
replace r2clock65=1 if r2clock_scorer>=0 & r2clock_scorer<=1

gen r2word65=0 if r2wordrecall0_20>3 & r2wordrecall0_20<=20
replace r2word65=1 if r2wordrecall0_20>=0 & r2wordrecall0_20<=3

gen r2datena65=0 if r2date_prvp>3 & r2date_prvp<=8
replace r2datena65=1 if r2date_prvp>=0 & r2date_prvp<=3
```

```
*CREATE COGNITIVE DOMAIN SCORE*
gen r2domain65 = r2clock65+r2word65+r2datena65

*UPDATE COGNITIVE CLASSIFICATION*
*PROBABLE DEMENTIA*
replace r2demclas=1 if r2demclas==. & (cg2speaktosp==1 | cg2speaktosp==-1) & (r2domain65==2 |
r2domain65==3)
*POSSIBLE DEMENTIA*
replace r2demclas=2 if r2demclas==. & (cg2speaktosp==1 | cg2speaktosp==-1) & r2domain65==1
*NO DEMENTIA*
replace r2demclas=3 if r2demclas==. & (cg2speaktosp==1 | cg2speaktosp==-1) & r2domain65==0

*Label variables and values*
label variable r2ad8_dem "Dementia classification based on proxy AD8 report"
label define r2ad8_dem_values 1 "1 Meets dementia criteria" 2 "2 Does not meet dementia criteria"
label values r2ad8_dem r2ad8_dem_values
label variable r2demclas "R2 NHATS Dementia Diagnosis 65+"
label define dementialabel652 1 "1 Probable dementia" 2 "2 Possible dementia" 3 "3 No dementia" -1 "-
1 Deceased or nursing home resident in R1 and R2" -9 "-9 Missing"
label values r2demclas dementialabel652
label define domain_labels2 0 "0 Does not meet criteria" 1 "1 Meets criteria"
label values r2clock65 r2word65 r2datena65 domain_labels2
label define domain65_label2 0 "0 Not impaired" 1 "Impaired in 1 domain" 2 "Impaired in 2 domains" 3
"Impaired in 3 domains"
label values r2domain65 domain65_labels2
tab r2demclas
```