```
1  ********************************************************************************************************************************
   ************************************************************
2  ***SYNTAX FOR "Socio-economic status and trajectories of a novel multidimensional metric of Active
   and Healthy Ageing: the English Longitudinal Study of Ageing"***
3  ********************************************************************************************************************************
   ************************************************************
4
5  * STATA version: 17.0, BE-Basic Edition
6
7  * STATA citation: StataCorp. 2021. Stata Statistical Software: Release 17. College Station, TX:
   StataCorp LLC.
8
9  * Data citation: Banks, J., Batty, G. David, Breedvelt, J., Coughlin, K., Crawford, R., Marmot, M.,
   Nazroo, J., Oldfield, Z., Steel, N., Steptoe, A., Wood, M., Zaninotto, P. (2021). English
   Longitudinal Study of Ageing: Waves 0-9, 1998-2019. [data collection]. 37th Edition. UK Data
   Service. SN: 5050, DOI: 10.5255/UKDA-SN-5050-24
10
11 * Data access statement: ELSA data from all waves are available through the UK Data Service
   (https://ukdataservice.ac.uk/). The main ELSA dataset is safeguarded and can be accessed via
   https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=5050#!/access-data. More information
   on how to access ELSA, including the conditions of use, can be found on the UK Data Service website
   (https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=5050#!/details) and the ELSA
   website (https://www.elsa-project.ac.uk/accessing-elsa-data).
12
13 * Date of data access/download (dd/mm/yyyy): 25/05/2022
14
15 * Project ID: 212810
16
17 * Data documentation: Documentation pertaining to ELSA (e.g., data dictionaries, questionnaires,
   technical reports, user guides) is available on the UK Data Service website
   (https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=5050#!/documentation) and the ELSA
   website (https://www.elsa-project.ac.uk/data-and-documentation).
18
19 **********************
20 ***DATA PROCESSING***
21 **********************
22
23 * Change working directory - add pathname in between quotation marks for Windows
24 cd ""
25
26 * Variables Wave 2
27 use idauniq Heill Helim MmAlone MmHSS Hehelf w2nssec8 MmTrya MMWlkA MmTryb MMWlkB heada01 heada02
   heada03 heada04 heada05 heada06 heada07 heada08 heada09 heada10 headb01 headb02 headb03 headb04
   headb05 headb06 headb07 headb08 headb09 headb10 headb11 headb12 headb13 scorg01 scorg02 scorg03
   scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scpt04 scpt05 sampsta dhager
   diagr indager DhSex DiSex indsex HeSmk HESka HeSkb HeSkc HeSkd HeSke HeSkf HeActa HeActb HeActc scako
    fqethnr scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm
   scqoln scqolo scqolp scqolq scqolr scqols PScedA PScedB PScedC PScedD PScedE PScedF PScedG PScedH
   scfeela scfeelb scfeelc CfDScr CfTest CfLisEn CfAni CfLisD MmWill MmSaf MmAvsp MmWala MmAid mmaidc
   CfDatD CfDatM CfDatY CfDay CfWrds MmSchs bhesmk bheska finstat w2wgt using wave_2_core_data_v4.dta
28 * Describe dataset
29 describe
30 * Sort from lowest to highest participant identifier (ID)
31 sort idauniq
32 * Rename variables to shorter or more convenient forms
33 rename MmSchs mmschs
34 rename MmWill mmwill
35 rename MmSaf mmsaf
36 rename MmAvsp mmavsp
37 rename MmWala mmwala
38 rename MmAid mmaid
39 rename CfDatD cfdatd
40 rename CfDatM cfdatm
```

```
41  rename CfDatY cfdaty
42  rename CfDay cfday
43  rename CfTest cftest
44  rename CfWrds cfwrds
45  rename Heill heill
46  rename Helim helim
47  rename MmAlone mmalone
48  rename MmHSS mmhss
49  rename Hehelf hehelf
50  rename MmTrya mmtrya
51  rename MMWlkA mmwlka
52  rename MmTryb mmtryb
53  rename MMWlkB mmwlkb
54  rename sampsta samptyp2
55  rename DhSex dhsex
56  rename DiSex disex
57  rename HeSmk hesmk
58  rename HESka heska
59  rename HeSkb heskb
60  rename HeSkc heskc
61  rename HeSkd heskd
62  rename HeSke heske
63  rename HeSkf heskf
64  rename HeActa heacta
65  rename HeActb heactb
66  rename HeActc heactc
67  rename PScedA psceda
68  rename PScedB pscedb
69  rename PScedC pscedc
70  rename PScedD pscedd
71  rename PScedE pscede
72  rename PScedF pscedf
73  rename PScedG pscedg
74  rename PScedH pscedh
75  rename CfDScr cfdscr
76  rename CfLisEn cflisen
77  rename CfAni cfani
78  rename CfLisD cflisd
79  rename w2nssec8 nssec8
80  rename finstat finstatw2
81  * Generate a new variable called wave and assign the number 2 to each observation (to designate Wave
    2)
82  gen wave = 2
83  * Save Wave 2 core dataset
84  save wave2.dta
85
86  * Variables Wave 3
87  use idauniq heill helim w3nssec8 mmalone mmhss hegenh mmtrya mmwlka mmtryb mmwlkb hemobwa hemobsi
    hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba
    headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo headl96 scorg01
    scorg02 scorg03 scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scpt04 scpt05
    sampsta dhager diagr indager dhsex disex indsex hesmk heska heskb heskc heskd heske heskf chesmk
    cheska heacta heactb heactc scako fqethnr w3edqual scqola scqolb scqolc scqold scqole scqolf scqolg
    scqolh scqoli scqolj scqolk scqoll scqolm scqoln scqolo scqolp scqolq scqolr scqols psceda pscedb
    pscedc pscedd pscede pscedf pscedg pscedh scfeela scfeelb scfeelc cfdscr cftest cflisen cfani cflisd
    mmwill mmsaf mmavsp mmwala mmaid mmaidc cfdatd cfdatm cfdaty cfday cfwrds mmschs finstat w3xwgt using
     wave_3_elsa_data_v4.dta
88  * Describe dataset
89  describe
90  * Sort from lowest to highest participant ID
91  sort idauniq
92  * Rename variables to ensure consistency across waves
93  rename w3nssec8 nssec8
```

```
rename sampsta samptyp3
rename w3edqual edqual
rename finstat finstatw3
* Generate a new variable called wave and assign the number 3 to each observation (to designate Wave 3)
gen wave = 3
* Save Wave 3 core dataset
save wave3.dta

* Variables Wave 4
use idauniq heill helim w4nssec8 mmalone mmhss hehelf mmtrya mmwlka mmtryb mmwlkb hemobwa hemobsi hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba headlea headlbe headlwc headlma headlpr headlsh headlte headlme headlho headlmo headl96 scorg01 scorg02 scorg03 scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scpt03 scpt04 samptyp diagr indager dhsex disex indsex hesmk heska heskb heskc heskd heske heskf dhesmk dheska heacta heactb heactc scako fqethnr w4edqual scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm scqoln scqolo scqolp scqolq scqolr scqols psceda pscedb pscedc pscedd pscede pscedf pscedg pscedh scfeela scfeelb scfeelc cfdscr cftest cflisen cfani cflisd mmwill mmsaf mmavsp mmwala mmaid mmaidc cfdatd cfdatm cfdaty cfday cfwrds mmschs finstat4 w4xwgt using wave_4_elsa_data_v3.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename w4nssec8 nssec8
rename headlte headlph
rename scpt04 scpt05
rename scpt03 scpt04
rename w4edqual edqual
rename samptyp samptyp4
rename finstat4 finstatw4
* Generate a new variable called wave and assign the number 4 to each observation (to designate Wave 4)
gen wave = 4
* Save Wave 4 core dataset
save wave4.dta

* Variables Wave 5
use idauniq heill helim mmalone w5nssec8 mmhss hehelf mmtrya mmwlka mmtryb mmwlkb hemobwa hemobsi hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba headlea headlbe headlwc headlma headlpr headlsh headlte headlme headlho headlmo headl96 scorg01 scorg02 scorg03 scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scpt03 scpt04 samptyp diagr indager dhsex disex indsex hesmk heska heskb heskc heskd heske heskf dhesmk dheska heacta heactb heactc scako fqethnr w5edqual scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm scqoln scqolo scqolp scqolq scqolr scqols psceda pscedb pscedc pscedd pscede pscedf pscedg pscedh scfeela scfeelb scfeelc cfdscr cftest cflisen cfani cflisd mmwill mmsaf mmavsp mmwala mmaid mmaidc cfdatd cfdatm cfdaty cfday cfwrds mmschs finstatw5 w5xwgt using wave_5_elsa_data_v4.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename w5nssec8 nssec8
rename headlte headlph
rename scpt04 scpt05
rename scpt03 scpt04
rename w5edqual edqual
rename samptyp samptyp5
* Generate a new variable called wave and assign the number 5 to each observation (to designate Wave 5)
gen wave = 5
```

```
* Save Wave 5 core dataset
save wave5.dta

* Variables Wave 6
use idauniq Heill Helim w6nssec8 MmAlone MmHSS Hehelf MmTrya MMWlkA MmTryb MMWlkB hemobwa hemobsi
hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba
headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo headl96 scorg01
scorg02 scorg03 scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scptr3 scptr4
samptyp indager DhSex DiSex indsex HeSmk HESka HeSkb HeSkc HeSkd HeSke HeSkf HeActa HeActb HeActc
scako Fqethnr scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll
scqolm scqoln scqolo scqolp scqolq scqolr scqols PScedA PScedB PScedC PScedD PScedE PScedF PScedG
PScedH scfeela scfeelb scfeelc CfDScr CfTest CfLisEn CfLisD MmWill MmSaf MmAvsp MmWala MmAid CfDatD
CfDatM CfDatY CfDay CfWrds MmSchs finstatw6 w6xwgt using wave_6_elsa_data_v2.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename MmSchs mmschs
rename MmWill mmwill
rename MmSaf mmsaf
rename MmAvsp mmavsp
rename MmWala mmwala
rename MmAid mmaid
rename CfDatD cfdatd
rename CfDatM cfdatm
rename CfDatY cfdaty
rename CfDay cfday
rename CfTest cftest
rename CfWrds cfwrds
rename w6nssec8 nssec8
rename Heill heill
rename Helim helim
rename MmAlone mmalone
rename MmHSS mmhss
rename Hehelf hehelf
rename MmTrya mmtrya
rename MMWlkA mmwlka
rename MmTryb mmtryb
rename MMWlkB mmwlkb
rename scptr3 scpt04
rename scptr4 scpt05
rename DhSex dhsex
rename DiSex disex
rename HeSmk hesmk
rename HESka heska
rename HeSkb heskb
rename HeSkc heskc
rename HeSkd heskd
rename HeSke heske
rename HeSkf heskf
rename HeActa heacta
rename HeActb heactb
rename HeActc heactc
rename Fqethnr fqethnr
rename samptyp samptyp6
rename PScedA psceda
rename PScedB pscedb
rename PScedC pscedc
rename PScedD pscedd
rename PScedE pscede
rename PScedF pscedf
rename PScedG pscedg
```

```
rename PScedH pscedh
rename CfDScr cfdscr
rename CfLisEn cflisen
rename CfLisD cflisd
* Generate a new variable called wave and assign the number 6 to each observation (to designate Wave 6)
gen wave = 6
* Save Wave 6 core dataset
save wave6.dta

* Variables Wave 7
use idauniq Heill Helim NSSEC MmAlone MmHSS Hehelf MmTrya MMWlkA MmTryb MMWlkB hemobwa hemobsi hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo headl96 scorg01 scorg02 scorg03 scorg04 scorg05 scorg06 scorg07 scorg08 scacta scactb scactc scactd scptr3 scptr4 samptyp indager DhSex DiSex indsex HeSmk HESka HeSkb HeSkc HeSkd HeSke HeSkf HeActa HeActb HeActc scako Fqethnr scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm scqoln scqolo scqolp scqolq scqolr scqols PScedA PScedB PScedC PScedD PScedE PScedF PScedG PScedH scfeela scfeelb scfeelc CfDScr CfTest CfLisEn CfAni CfLisD MmWill MmSaf MmAvsp MmWala MmAid CfDatD CfDatM CfDatY CfDay CfWrds MmSchs finstatw7 w7xwgt using wave_7_elsa_data.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename MmSchs mmschs
rename MmWill mmwill
rename MmSaf mmsaf
rename MmAvsp mmavsp
rename MmWala mmwala
rename MmAid mmaid
rename CfDatD cfdatd
rename CfDatM cfdatm
rename CfDatY cfdaty
rename CfDay cfday
rename CfTest cftest
rename CfWrds cfwrds
rename Heill heill
rename Helim helim
rename MmAlone mmalone
rename MmHSS mmhss
rename Hehelf hehelf
rename MmTrya mmtrya
rename MMWlkA mmwlka
rename MmTryb mmtryb
rename MMWlkB mmwlkb
rename scptr3 scpt04
rename scptr4 scpt05
rename DhSex dhsex
rename DiSex disex
rename HeSmk hesmk
rename HESka heska
rename HeSkb heskb
rename HeSkc heskc
rename HeSkd heskd
rename HeSke heske
rename HeSkf heskf
rename HeActa heacta
rename HeActb heactb
rename HeActc heactc
rename Fqethnr fqethnr
rename samptyp samptyp7
rename PScedA psceda
```

```
245  rename PScedB pscedb
246  rename PScedC pscedc
247  rename PScedD pscedd
248  rename PScedE pscede
249  rename PScedF pscedf
250  rename PScedG pscedg
251  rename PScedH pscedh
252  rename CfDScr cfdscr
253  rename CfLisEn cflisen
254  rename CfAni cfani
255  rename CfLisD cflisd
256  * Generate a new variable called wave and assign the number 7 to each observation (to designate Wave
     7)
257  gen wave = 7
258  * Save Wave 7 core dataset
259  save wave7.dta
260
261  * Variables Wave 8
262  use idauniq heill helim w8nssec8 mmalone mmhss hehelf mmtrya mmwlka mmtryb mmwlkb hemobwa hemobsi
     hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba
     headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo headl96 scorgpo
     scorgnw scorgrl scorgch scorged scorgsc scorgsp scorg95 scacta scactb scactc scactd scptruk scptrab
     samptyp indager indsex hesmk heska heskb heskc heskd heske heskf heacta heactb heactc scako fqethnmr
     scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm scqoln
     scqolo scqolp scqolq scqolr scqols psceda pscedb pscedc pscedd pscede pscedf pscedg pscedh scfeela
     scfeelb scfeelc cfdscr cftest cflisen cfani cflisd mmwill mmsaf mmavsp mmwala mmaid cfdatd cfdatm
     cfdaty cfday cfwrds mmschs finstat w8xwgt using wave_8_elsa_data_eul_v2.dta
263  * Describe dataset
264  describe
265  * Sort from lowest to highest participant ID
266  sort idauniq
267  * Rename variables to ensure consistency across waves
268  rename w8nssec8 nssec8
269  rename scorgpo scorg01
270  rename scorgnw scorg02
271  rename scorgrl scorg03
272  rename scorgch scorg04
273  rename scorged scorg05
274  rename scorgsc scorg06
275  rename scorgsp scorg07
276  rename scorg95 scorg08
277  rename scptruk scpt04
278  rename scptrab scpt05
279  rename fqethnmr fqethnr
280  rename samptyp samptyp8
281  rename finstat finstatw8
282  * Generate a new variable called wave and assign the number 8 to each observation (to designate Wave
     8)
283  gen wave = 8
284  * Save Wave 8 core dataset
285  save wave8.dta
286
287  * Variables Wave 9
288  use idauniq heill helim w9nssec8 mmalone mmhss hehelf mmtrya mmwlka mmtryb mmwlkb hemobwa hemobsi
     hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi hemob96 headldr headlwa headlba
     headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo headl96 scorgpo
     scorgnw scorgrl scorgch scorged scorgsc scorgsp scorg95 scacta scactb scactc scactd scptruk scptrab
     samptyp indager indsex hesmk heska heskb heskc heskd heske heskf heacta heactb heactc scalcm fqethnmr
      scqola scqolb scqolc scqold scqole scqolf scqolg scqolh scqoli scqolj scqolk scqoll scqolm scqoln
     scqolo scqolp scqolq scqolr scqols psceda pscedb pscedc pscedd pscede pscedf pscedg pscedh scfeela
     scfeelb scfeelc cfdscr cftest cflisen cfani cflisd mmwill mmsaf mmavsp mmwala mmaid cfdatd cfdatm
     cfdaty cfday cfwrds mmschs finstat w9xwgt using wave_9_elsa_data_eul_v1.dta
289  * Describe dataset
```

```
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename w9nssec8 nssec8
rename scorgpo scorg01
rename scorgnw scorg02
rename scorgrl scorg03
rename scorgch scorg04
rename scorged scorg05
rename scorgsc scorg06
rename scorgsp scorg07
rename scorg95 scorg08
rename scptruk scpt04
rename scptrab scpt05
rename scalcm scako
rename fqethnmr fqethnr
rename samptyp samptyp9
rename finstat finstatw9
* Generate a new variable called wave and assign the number 9 to each observation (to designate Wave 9)
gen wave = 9
* Save Wave 9 core dataset
save wave9.dta

* Variables Wave 2 Financial Derived
use idauniq totwq5_bu_s using wave_2_financial_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 2 financial derived dataset
save wave2financial.dta

* Variables Wave 2 Nurse
use idauniq mmssre mmssti mmstre mmstti mmftre2 mmrrre mmftti mmrrfti mmssna mmstna mmftna mmcrre mmcrna mmrrna cfib hdl trig ldl hscrp hba1c mmgsd1 mmgsn1 mmgsd2 mmgsn2 mmgsd3 mmgsn3 mmgswil mmgsdom mmgssta mmgstp mmgsres mmbcsc mmsssc mmstsc mmftsc mmcrav mmcrsc mmrrsc mmrrtti mmrroc using wave_2_nurse_data_v2.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 2 nurse dataset
save wave2nurse.dta

* Variables Wave 2 Derived
use idauniq w2edqual using wave_2_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to shorter or more convenient forms
rename w2edqual edqual
* Save Wave 2 derived dataset
save wave2derived.dta

* Variables Wave 3 Financial Derived
use idauniq totwq5_bu_s using wave_3_financial_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
```

```
* Save Wave 3 financial derived dataset
save wave3financial.dta

* Variables Wave 4 Financial Derived
use idauniq totwq5_bu_s using wave_4_financial_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 4 financial derived dataset
save wave4financial.dta

* Variables Wave 4 Nurse
use idauniq mmssre mmssti mmstre mmstti mmftre2 mmrrre mmftti mmrrfti mmssna mmstna mmftna mmcrre
mmcrna mmrrna cfib hdl trig ldl hscrp hba1c mmgsd1 mmgsn1 mmgsd2 mmgsn2 mmgsd3 mmgsn3 mmgswil mmgsdom
 mmgssta mmgstp mmgsres mmbcsc mmsssc mmstsc mmftsc mmcrav mmcrsc mmrrsc mmrrtti mmrroc using
wave_4_nurse_data.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 4 nurse dataset
save wave4nurse.dta

* Variables Wave 5 Financial Derived
use idauniq totwq5_bu_s using wave_5_financial_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 5 financial derived dataset
save wave5financial.dta

* Variables Wave 6 Financial Derived
use idauniq totwq5_bu_s using wave_6_financial_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 6 financial derived dataset
save wave6financial.dta

* Variables Wave 6 Nurse
use idauniq mmssre mmssti mmstre mmstti MmFTRe MMFTRE2 mmrrre mmftti mmrrfti mmssna mmstna mmftna
mmcrre mmcrna mmrrna cfib hdl trig ldl hscrp hba1c mmgsd1 mmgsn1 mmgsd2 mmgsn2 mmgsd3 mmgsn3 mmgswil
mmgsdom mmgssta mmgstp mmgsres mmbcsc mmsssc mmstsc mmftsc mmcrav mmcrsc mmrrsc mmrrtti MMRROC using
wave_6_elsa_nurse_data_v2.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Rename variables to ensure consistency across waves
rename MMRROC mmrroc
rename MmFTRe mmftre
rename MMFTRE2 mmftre2
* Save Wave 6 nurse dataset
save wave6nurse.dta

* Variables Wave 6 Derived
use idauniq edqual using wave_6_ifs_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
```

```
406  sort idauniq
407  * Save Wave 6 derived dataset
408  save wave6derived.dta
409
410  * Variables Wave 7 Financial Derived
411  use idauniq totwq5_bu_s using wave_7_financial_derived_variables.dta
412  * Describe dataset
413  describe
414  * Sort from lowest to highest participant ID
415  sort idauniq
416  * Save Wave 7 financial derived dataset
417  save wave7financial.dta
418
419  * Variables Wave 7 Derived
420  use idauniq edqual using wave_7_ifs_derived_variables.dta
421  * Describe dataset
422  describe
423  * Sort from lowest to highest participant ID
424  sort idauniq
425  * Save Wave 7 derived dataset
426  save wave7derived.dta
427
428  * Variables Wave 8 Financial Derived
429  use idauniq totwq5_bu_s using wave_8_elsa_financial_dvs_eul_v1.dta
430  * Describe dataset
431  describe
432  * Sort from lowest to highest participant ID
433  sort idauniq
434  * Save Wave 8 financial derived dataset
435  save wave8financial.dta
436
437  * Variables Wave 8 Derived
438  use idauniq edqual using wave_8_elsa_ifs_dvs_eul_v1.dta
439  * Describe dataset
440  describe
441  * Sort from lowest to highest participant ID
442  sort idauniq
443  * Save Wave 8 derived dataset
444  save wave8derived.dta
445
446  * Variables Wave 8-9 Nurse
447  use idauniq wave cfib hdl trig ldl hscrp hba1c mmgsd1 mmgsn1 mmgsd2 mmgsn2 mmgsd3 mmgsn3 mmgswil
     mmgsdom mmgssta mmgstp mmgsres using elsa_nurse_w8w9_data_eul.dta
448  * Describe dataset
449  describe
450  * Sort from lowest to highest participant ID
451  sort idauniq
452  * Save Wave 8-9 nurse dataset
453  save wave89nurse.dta
454  * Keep data from Wave 8 only
455  keep if wave==8
456  * Save Wave 8 nurse dataset
457  save wave8nurse.dta
458  * Use Wave 8-9 nurse dataset
459  use wave89nurse.dta
460  * Keep data from Wave 9 only
461  keep if wave==9
462  * Save Wave 9 nurse dataset
463  save wave9nurse.dta
464
465  * Variables Wave 9 Financial Derived
466  use idauniq totwq5_bu_s using wave_9_financial_derived_variables.dta
467  * Describe dataset
```

```
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 9 financial derived dataset
save wave9financial.dta

* Wave 9 Derived variables
use idauniq edqual using wave_9_ifs_derived_variables.dta
* Describe dataset
describe
* Sort from lowest to highest participant ID
sort idauniq
* Save Wave 9 derived dataset
save wave9derived.dta

* Wave 2 complete data
* Merge core, financial, nurse, and derived datasets for Wave 2 using the participant ID
* Use Wave 2 core dataset
use wave2.dta
* One-to-one merge of data in memory with wave2financial.dta on participant ID
merge 1:1 idauniq using wave2financial.dta, generate (merge_financial2)
* Overwrite Wave 2 dataset, by replacing the previously saved file
save wave2.dta, replace
* Use the newly saved file for Wave 2
use wave2.dta
* One-to-one merge of data in memory with wave2nurse.dta on participant ID
merge 1:1 idauniq using wave2nurse.dta, generate (merge_nurse2)
* Overwrite Wave 2 dataset, by replacing the previously saved file
save wave2.dta, replace
* Use the newly saved file for Wave 2
use wave2.dta
* One-to-one merge of data in memory with wave2derived.dta on participant ID
merge 1:1 idauniq using wave2derived.dta, generate (merge_derived2)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 2 dataset, by replacing the previously saved file
save wave2.dta, replace

* Wave 3 complete data
* Merge core and financial datasets for Wave 3 using the participant ID
* Use Wave 3 core dataset
use wave3.dta
* One-to-one merge of data in memory with wave3financial.dta on participant ID
merge 1:1 idauniq using wave3financial.dta, generate (merge_financial3)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 3 dataset, by replacing the previously saved file
save wave3.dta, replace

* Wave 4 complete data
* Merge core, financial, and nurse datasets for Wave 4 using the participant ID
* Use Wave 4 core dataset
use wave4.dta
* One-to-one merge of data in memory with wave4financial.dta on participant ID
merge 1:1 idauniq using wave4financial.dta, generate (merge_financial4)
* Overwrite Wave 4 dataset, by replacing the previously saved file
save wave4.dta, replace
* Use the newly saved file for Wave 4
use wave4.dta
* One-to-one merge of data in memory with wave4nurse.dta on participant ID
merge 1:1 idauniq using wave4nurse.dta, generate (merge_nurse4)
* Sort from lowest to highest participant ID
sort idauniq
```

```
531  * Overwrite Wave 4 dataset, by replacing the previously saved file
532  save wave4.dta, replace
533
534  * Wave 5 complete data
535  * Merge core and financial datasets for Wave 5 using the participant ID
536  * Use Wave 5 core dataset
537  use wave5.dta
538  * One-to-one merge of data in memory with wave5financial.dta on participant ID
539  merge 1:1 idauniq using wave5financial.dta, generate (merge_financial5)
540  * Sort from lowest to highest participant ID
541  sort idauniq
542  * Overwrite Wave 5 dataset, by replacing the previously saved file
543  save wave5.dta, replace
544
545  * Wave 6 complete data
546  * Merge core, financial, nurse, and derived datasets for Wave 6 using the participant ID
547  * Use Wave 6 core dataset
548  use wave6.dta
549  * One-to-one merge of data in memory with wave6financial.dta on participant ID
550  merge 1:1 idauniq using wave6financial.dta, generate (merge_financial6)
551  * Overwrite Wave 6 dataset, by replacing the previously saved file
552  save wave6.dta, replace
553  * Use the newly saved file for Wave 6
554  use wave6.dta
555  * One-to-one merge of data in memory with wave6nurse.dta on participant ID
556  merge 1:1 idauniq using wave6nurse.dta, generate (merge_nurse6)
557  * Overwrite Wave 6 dataset, by replacing the previously saved file
558  save wave6.dta, replace
559  * Use the newly saved file for Wave 6
560  use wave6.dta
561  * One-to-one merge of data in memory with wave6derived.dta on participant ID
562  merge 1:1 idauniq using wave6derived.dta, generate (merge_derived6)
563  * Sort from lowest to highest participant ID
564  sort idauniq
565  * Overwrite Wave 6 dataset, by replacing the previously saved file
566  save wave6.dta, replace
567
568  * Wave 7 complete data
569  * Merge core, financial, and derived datasets for Wave 7 using the participant ID
570  * Use Wave 7 core dataset
571  use wave7.dta
572  * One-to-one merge of data in memory with wave7financial.dta on participant ID
573  merge 1:1 idauniq using wave7financial.dta, generate (merge_financial7)
574  * Overwrite Wave 7 dataset, by replacing the previously saved file
575  save wave7.dta, replace
576  * Use the newly saved file for Wave 7
577  use wave7.dta
578  * One-to-one merge of data in memory with wave7derived.dta on participant ID
579  merge 1:1 idauniq using wave7derived.dta, generate (merge_derived7)
580  * Sort from lowest to highest participant ID
581  sort idauniq
582  * Overwrite Wave 7 dataset, by replacing the previously saved file
583  save wave7.dta, replace
584
585  * Wave 8 complete data
586  * Merge core, financial, nurse, and derived datasets for Wave 8 using the participant ID
587  * Use Wave 8 core dataset
588  use wave8.dta
589  * One-to-one merge of data in memory with wave8financial.dta on participant ID
590  merge 1:1 idauniq using wave8financial.dta, generate (merge_financial8)
591  * Overwrite Wave 8 dataset, by replacing the previously saved file
592  save wave8.dta, replace
593  * Use the newly saved file for Wave 8
```

```
use wave8.dta
* One-to-one merge of data in memory with wave8nurse.dta on participant ID
merge 1:1 idauniq using wave8nurse.dta, generate (merge_nurse8)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 8 dataset, by replacing the previously saved file
save wave8.dta, replace
* Use the newly saved file for Wave 8
use wave8.dta
* One-to-one merge of data in memory with wave8derived.dta on participant ID
merge 1:1 idauniq using wave8derived.dta, generate (merge_derived8)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 8 dataset, by replacing the previously saved file
save wave8.dta, replace

* Wave 9 complete data
* Merge core, financial, nurse, and derived datasets for Wave 9 using the participant ID
* Use Wave 9 core dataset
use wave9.dta
* One-to-one merge of data in memory with wave9financial.dta on participant ID
merge 1:1 idauniq using wave9financial.dta, generate (merge_financial9)
* Overwrite Wave 9 dataset, by replacing the previously saved file
save wave9.dta, replace
* Use the newly saved file for Wave 9
use wave9.dta
* One-to-one merge of data in memory with wave9derived.dta on participant ID
merge 1:1 idauniq using wave9derived.dta, generate (merge_derived9)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 9 dataset, by replacing the previously saved file
save wave9.dta, replace
* Use the newly saved file for Wave 9
use wave9.dta
* One-to-one merge of data in memory with wave9nurse.dta on participant ID
merge 1:1 idauniq using wave9nurse.dta, generate (merge_nurse9)
* Sort from lowest to highest participant ID
sort idauniq
* Overwrite Wave 9 dataset, by replacing the previously saved file
save wave9.dta, replace

* Append Wave 3 dataset to Wave 2 dataset
use wave2.dta
append using wave3.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 4 dataset
append using wave4.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 5 dataset
append using wave5.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 6 dataset
append using wave6.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 7 dataset
append using wave7.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 8 dataset
```

```
append using wave8.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Append Wave 9 dataset
append using wave9.dta
* Sort by participant ID and wave (lowest to highest)
sort idauniq wave
* Assign a number in ascending order to each row of observations
gen ascnr = _n

* Organising dataset
* Generate a variable that assigns the observation number (i.e., 1 for first data collection timepoint, 2 for second data collection timepoint etc.) to each row by participant ID
bysort idauniq (wave): gen obsnr = _n
* Generate a variable that assigns the number of total observations to each row of data for a given participant
bysort idauniq: gen obscount = _N
* Check how many participants have data at 1 to 8 timepoints - the "if obsnr==1" statement is used to prevent participants with data at more than one timepoint from contributing to the counts more than once
tabulate obscount if obsnr==1
* Generate a variable that assigns the number 1 to the row representing participants' first observation
bysort idauniq (wave): gen first = 1 if _n==1
* Generate a variable that assigns the number 1 to the row representing participants' last observation
bysort idauniq (wave): gen last = 1 if _n==_N
* Generate a variable that assigns the number 1 to the row representing participants' first observation if this corresponds to Wave 2 (baseline)
bysort idauniq (wave): gen firstwave = 1 if obsnr==1 & wave==2
* Carry the value of this last variable forwards to the remainder of a participant's observations
bysort idauniq: gen firstwave_cons = firstwave[1]
* Install unique command
ssc install unique
* Count total number of participants and observations
unique idauniq
* 19,807 individuals, 80,750 observations
* Save dataset with a new name
save raw.dta

* Keep if participant is a core member (include core members who had a proxy or partial interview or those who had been interviewed in an institution)
keep if finstatw2=="C1CM" | inlist(finstatw3,"C1CM","C3CM") | inlist(finstatw4,"C1CM","C3CM","C4CM") | inlist(finstatw5,"C1CM","C3CM","C4CM") | inlist(finstatw6,1,7,14,25) | inlist(finstatw7,1,7,14,25,33) | inlist(finstatw8,1,7,14,25,33) | inlist(finstatw9,1,7,14,25,33,48)
* Count total number of participants and observations
unique idauniq
* 15,022 individuals, 68,496 observations
* Replace age = 90 if participant is aged 90+ years (collapsed in ELSA and coded as 99 at Wave 2, 3, and 4)
replace indager = 90 if indager==99 & inlist(wave,2,3,4)
* Replace age = 90 if participant is aged 90+ years (collapsed in ELSA and coded as -7 at Wave 5, 6, 7, 8, and 9)
replace indager = 90 if indager==-7 & inlist(wave,5,6,7,8,9)
* Save dataset with a new name
save data.dta

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b37-b40] Socio-cultural trips (reversed)
* [b7-b8] Holidays
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
* [b37] How often respondent goes to the cinema
replace scacta = . if scacta<0
* [b38] How often respondent eats out of the house
```

```
replace scactb = . if scactb<0
* [b39] How often respondent goes to an art gallery or museum
replace scactc = . if scactc<0
* [b40] How often respondent goes to the theatre, a concert, or the opera
replace scactd = . if scactd<0
* [b7] Respondent has taken a holiday in the UK in the last 12 months
replace scpt04 = . if scpt04<0
* [b8] Respondent has taken a holiday abroad in the last 12 months
replace scpt05 = . if scpt05<0

* Reverse the negatively framed variables (this creates new variables and adds the "rev" prefix to the original variable names)
revrs scacta
revrs scactb
revrs scactc
revrs scactd
* Collapse categories with a small number of participants
replace revscacta = 5 if revscacta == 6
replace revscactc = 5 if revscactc == 6
replace revscactd = 5 if revscactd == 6
replace revscactb = 1 if revscactb ==2
replace revscactb = 2 if revscactb ==3
replace revscactb = 3 if revscactb ==4
replace revscactb = 4 if revscactb ==5
replace revscactb = 5 if revscactb ==6

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b26-b36] (+ 2 omitted) ADL and IADL disabilities (reversed)
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace heada01 = . if heada01<0
replace heada02 = . if heada02<0
replace heada03 = . if heada03<0
replace heada04 = . if heada04<0
replace heada05 = . if heada05<0
replace heada06 = . if heada06<0
replace heada07 = . if heada07<0
replace heada08 = . if heada08<0
replace heada09 = . if heada09<0
replace heada10 = . if heada10<0
replace headb01 = . if headb01<0
replace headb02 = . if headb02<0
replace headb03 = . if headb03<0
replace headb04 = . if headb04<0
replace headb05 = . if headb05<0
replace headb06 = . if headb06<0
replace headb07 = . if headb07<0
replace headb08 = . if headb08<0
replace headb09 = . if headb09<0
replace headb10 = . if headb10<0
replace headb11 = . if headb11<0
replace headb12 = . if headb12<0
replace headb13 = . if headb13<0

replace headldr = . if headldr<0
replace headlwa = . if headlwa<0
replace headlba = . if headlba<0
replace headlea = . if headlea<0
replace headlbe = . if headlbe<0
replace headlwc = . if headlwc<0
replace headlma = . if headlma<0
replace headlpr = . if headlpr<0
replace headlsh = . if headlsh<0
replace headlph = . if headlph<0
```

```
replace headlme = . if headlme<0
replace headlho = . if headlho<0
replace headlmo = . if headlmo<0
replace headl96 = . if headl96<0

* ADL
* [b26] ADL: difficulty dressing, including putting on shoes and socks
replace headldr = 2 if headldr == 0
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace headldr = 0 if headldr == 1
* Assign the number 1 if the participant reported no difficulties performing the first listed activity
replace headldr = 1 if headldr == 2
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace headldr = 0 if (headb01 == 1 | headb02 == 1 | headb03 == 1 | headb04 == 1 | headb05 == 1 |
headb06 == 1 | headb07 == 1 | headb08 == 1 | headb09 == 1 | headb10 == 1 | headb11 == 1 | headb12 ==
1 | headb13 == 1) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headldr = 1 if inlist(headb01,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb02,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb03,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb04,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb05,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb06,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb07,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb08,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb09,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb10,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb11,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb12,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
replace headldr = 1 if inlist(headb13,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0

* [b27] ADL: difficulty walking across a room
replace headlwa = 2 if headlwa == 0
* Assign the number 0 if the participant reported difficulties performing the second listed activity
replace headlwa = 0 if headlwa == 1
* Assign the number 1 if the participant reported no difficulties performing the second listed
activity
replace headlwa = 1 if headlwa == 2
* Assign the number 0 if the participant reported difficulties performing the second listed activity
replace headlwa = 0 if (headb01 == 2 | headb02 == 2 | headb03 == 2 | headb04 == 2 | headb05 == 2 |
headb06 == 2 | headb07 == 2 | headb08 == 2 | headb09 == 2 | headb10 == 2 | headb11 == 2 | headb12 ==
2 | headb13 == 2) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlwa = 1 if inlist(headb01,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb02,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb03,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb04,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb05,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb06,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb07,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb08,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb09,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb10,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb11,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb12,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
replace headlwa = 1 if inlist(headb13,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0

* [b28] ADL: difficulty bathing or showering
replace headlba = 2 if headlba == 0
* Assign the number 0 if the participant reported difficulties performing the third listed activity
replace headlba = 0 if headlba == 1
```

```
* Assign the number 1 if the participant reported no difficulties performing the third listed activity
replace headlba = 1 if headlba == 2
* Assign the number 0 if the participant reported difficulties performing the third listed activity
replace headlba = 0 if (headb01 == 3 | headb02 == 3 | headb03 == 3 | headb04 == 3 | headb05 == 3 |
headb06 == 3 | headb07 == 3 | headb08 == 3 | headb09 == 3 | headb10 == 3 | headb11 == 3 | headb12 ==
3 | headb13 == 3) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlba = 1 if inlist(headb01,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb02,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb03,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb04,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb05,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb06,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb07,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb08,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb09,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb10,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb11,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb12,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
replace headlba = 1 if inlist(headb13,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0

* [b29] ADL: difficulty eating, such as cutting up your food
replace headlea = 2 if headlea == 0
* Assign the number 0 if the participant reported difficulties performing the fourth listed activity
replace headlea = 0 if headlea == 1
* Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity
replace headlea = 1 if headlea == 2
* Assign the number 0 if the participant reported difficulties performing the fourth listed activity
replace headlea = 0 if (headb01 == 4 | headb02 == 4 | headb03 == 4 | headb04 == 4 | headb05 == 4 |
headb06 == 4 | headb07 == 4 | headb08 == 4 | headb09 == 4 | headb10 == 4 | headb11 == 4 | headb12 ==
4 | headb13 == 4) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlea = 1 if inlist(headb01,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb02,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb03,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb04,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb05,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb06,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb07,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb08,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb09,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb10,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb11,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb12,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
replace headlea = 1 if inlist(headb13,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0

* [b30] ADL: difficulty getting in or out of bed
replace headlbe = 2 if headlbe == 0
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace headlbe = 0 if headlbe == 1
* Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
replace headlbe = 1 if headlbe == 2
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace headlbe = 0 if (headb01 == 5 | headb02 == 5 | headb03 == 5 | headb04 == 5 | headb05 == 5 |
headb06 == 5 | headb07 == 5 | headb08 == 5 | headb09 == 5 | headb10 == 5 | headb11 == 5 | headb12 ==
5 | headb13 == 5) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlbe = 1 if inlist(headb01,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb02,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
```

```
replace headlbe = 1 if inlist(headb03,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb04,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb05,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb06,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb07,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb08,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb09,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb10,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb11,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb12,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
replace headlbe = 1 if inlist(headb13,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0

* [b31] ADL: difficulty using the toilet, including getting up or down
replace headlwc = 2 if headlwc == 0
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace headlwc = 0 if headlwc == 1
* Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
replace headlwc = 1 if headlwc == 2
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace headlwc = 0 if (headb01 == 6 | headb02 == 6 | headb03 == 6 | headb04 == 6 | headb05 == 6 |
headb06 == 6 | headb07 == 6 | headb08 == 6 | headb09 == 6 | headb10 == 6 | headb11 == 6 | headb12 ==
6 | headb13 == 6) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlwc = 1 if inlist(headb01,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb02,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb03,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb04,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb05,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb06,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb07,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb08,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb09,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb10,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb11,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb12,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
replace headlwc = 1 if inlist(headb13,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0

* IADL
* [b32] IADL: difficulty using a map to figure out how to get around in a strange place
replace headlma = 2 if headlma == 0
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace headlma = 0 if headlma == 1
* Assign the number 1 if the participant reported no difficulties performing the first listed activity
replace headlma = 1 if headlma == 2
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace headlma = 0 if (headb01 == 7 | headb02 == 7 | headb03 == 7 | headb04 == 7 | headb05 == 7 |
headb06 == 7 | headb07 == 7 | headb08 == 7 | headb09 == 7 | headb10 == 7 | headb11 == 7 | headb12 ==
7 | headb13 == 7) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlma = 1 if inlist(headb01,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb02,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb03,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb04,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb05,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb06,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb07,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb08,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb09,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb10,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb11,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
replace headlma = 1 if inlist(headb12,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
```

```
replace headlma = 1 if inlist(headb13,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0

* [b33] IADL: difficulty preparing a hot meal
replace headlpr = 2 if headlpr == 0
* Assign the number 0 if the participant reported difficulties performing the second listed activity
replace headlpr = 0 if headlpr == 1
* Assign the number 1 if the participant reported no difficulties performing the second listed activity
replace headlpr = 1 if headlpr == 2
* Assign the number 0 if the participant reported difficulties performing the second listed activity
replace headlpr = 0 if (headb01 == 8 | headb02 == 8 | headb03 == 8 | headb04 == 8 | headb05 == 8 | headb06 == 8 | headb07 == 8 | headb08 == 8 | headb09 == 8 | headb10 == 8 | headb11 == 8 | headb12 == 8 | headb13 == 8) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in headb01-headb13 and data are not missing
replace headlpr = 1 if inlist(headb01,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb02,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb03,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb04,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb05,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb06,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb07,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb08,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb09,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb10,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb11,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb12,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
replace headlpr = 1 if inlist(headb13,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0

* [b34] IADL: difficulty shopping for groceries
replace headlsh = 2 if headlsh == 0
* Assign the number 0 if the participant reported difficulties performing the third listed activity
replace headlsh = 0 if headlsh == 1
* Assign the number 1 if the participant reported no difficulties performing the third listed activity
replace headlsh = 1 if headlsh == 2
* Assign the number 0 if the participant reported difficulties performing the third listed activity
replace headlsh = 0 if (headb01 == 9 | headb02 == 9 | headb03 == 9 | headb04 == 9 | headb05 == 9 | headb06 == 9 | headb07 == 9 | headb08 == 9 | headb09 == 9 | headb10 == 9 | headb11 == 9 | headb12 == 9 | headb13 == 9) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in headb01-headb13 and data are not missing
replace headlsh = 1 if inlist(headb01,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb02,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb03,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb04,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb05,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb06,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb07,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb08,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb09,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb10,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb11,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb12,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
replace headlsh = 1 if inlist(headb13,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0

* [b35] IADL: difficulty making telephone calls
replace headlph = 2 if headlph == 0
* Assign the number 0 if the participant reported difficulties performing the fourth listed activity
replace headlph = 0 if headlph == 1
* Assign the number 1 if the participant reported no difficulties performing the fourth listed activity
replace headlph = 1 if headlph == 2
* Assign the number 0 if the participant reported difficulties performing the fourth listed activity
```

```
replace headlph = 0 if (headb01 == 10 | headb02 == 10 | headb03 == 10 | headb04 == 10 | headb05 == 10 | headb06 == 10 | headb07 == 10 | headb08 == 10 | headb09 == 10 | headb10 == 10 | headb11 == 10 | headb12 == 10 | headb13 == 10) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in headb01-headb13 and data are not missing
replace headlph = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
replace headlph = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0

* (1 omitted) IADL: difficulty talking medications
replace headlme = 2 if headlme == 0
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace headlme = 0 if headlme == 1
* Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
replace headlme = 1 if headlme == 2
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace headlme = 0 if (headb01 == 11 | headb02 == 11 | headb03 == 11 | headb04 == 11 | headb05 == 11 | headb06 == 11 | headb07 == 11 | headb08 == 11 | headb09 == 11 | headb10 == 11 | headb11 == 11 | headb12 == 11 | headb13 == 11) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in headb01-headb13 and data are not missing
replace headlme = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
replace headlme = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0

* [b36] IADL: difficulty doing work around the house or garden
replace headlho = 2 if headlho == 0
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace headlho = 0 if headlho == 1
* Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
replace headlho = 1 if headlho == 2
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace headlho = 0 if (headb01 == 12 | headb02 == 12 | headb03 == 12 | headb04 == 12 | headb05 == 12 | headb06 == 12 | headb07 == 12 | headb08 == 12 | headb09 == 12 | headb10 == 12 | headb11 == 12 | headb12 == 12 | headb13 == 12) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in headb01-headb13 and data are not missing
replace headlho = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
```

```
replace headlho = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
replace headlho = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0

* (1 omitted) IADL: difficulty managing money, such as paying bills and keeping track of expenses
replace headlmo = 2 if headlmo == 0
* Assign the number 0 if the participant reported difficulties performing the seventh listed activity
replace headlmo = 0 if headlmo == 1
* Assign the number 1 if the participant reported no difficulties performing the seventh listed
activity
replace headlmo = 1 if headlmo == 2
* Assign the number 0 if the participant reported difficulties performing the seventh listed activity
replace headlmo = 0 if (headb01 == 13 | headb02 == 13 | headb03 == 13 | headb04 == 13 | headb05 == 13
 | headb06 == 13 | headb07 == 13 | headb08 == 13 | headb09 == 13 | headb10 == 13 | headb11 == 13 |
headb12 == 13 | headb13 == 13) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
replace headlmo = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
replace headlmo = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0

* Save dataset with a new name
save datavariables.dta

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b16-b25] Mobility (reversed)
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace hemobwa = . if hemobwa<0
replace hemobsi = . if hemobsi<0
replace hemobch = . if hemobch<0
replace hemobcs = . if hemobcs<0
replace hemobcl = . if hemobcl<0
replace hemobst = . if hemobst<0
replace hemobre = . if hemobre<0
replace hemobpu = . if hemobpu<0
replace hemobli = . if hemobli<0
replace hemobpi = . if hemobpi<0
replace hemob96 = . if hemob96<0

* [b16] Mobility: difficulty walking 100 yards
replace hemobwa = 2 if hemobwa == 0
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace hemobwa = 0 if hemobwa == 1
* Assign the number 1 if the participant reported no difficulties performing the first listed activity
replace hemobwa = 1 if hemobwa == 2
* Assign the number 0 if the participant reported difficulties performing the first listed activity
replace hemobwa = 0 if (heada01 == 1 | heada02 == 1 | heada03 == 1 | heada04 == 1 | heada05 == 1 |
heada06 == 1 | heada07 == 1 | heada08 == 1 | heada09 == 1 | heada10 == 1) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
```

```
    heada01-heada10 and data are not missing
1104  replace hemobwa = 1 if inlist(heada01,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1105  replace hemobwa = 1 if inlist(heada02,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1106  replace hemobwa = 1 if inlist(heada03,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1107  replace hemobwa = 1 if inlist(heada04,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1108  replace hemobwa = 1 if inlist(heada05,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1109  replace hemobwa = 1 if inlist(heada06,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1110  replace hemobwa = 1 if inlist(heada07,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1111  replace hemobwa = 1 if inlist(heada08,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1112  replace hemobwa = 1 if inlist(heada09,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1113  replace hemobwa = 1 if inlist(heada10,2,3,4,5,6,7,8,9,10,96) & hemobwa!=0
1114
1115  * [b17] Mobility: difficulty sitting for about two hours
1116  replace hemobsi = 2 if hemobsi == 0
1117  * Assign the number 0 if the participant reported difficulties performing the second listed activity
1118  replace hemobsi = 0 if hemobsi == 1
1119  * Assign the number 1 if the participant reported no difficulties performing the second listed
    activity
1120  replace hemobsi = 1 if hemobsi == 2
1121  * Assign the number 0 if the participant reported difficulties performing the second listed activity
1122  replace hemobsi = 0 if (heada01 == 2 | heada02 == 2 | heada03 == 2 | heada04 == 2 | heada05 == 2 |
    heada06 == 2 | heada07 == 2 | heada08 == 2 | heada09 == 2 | heada10 == 2) & wave==2
1123  * Assign the number 1 if the participant reported any answer other than the listed activity in
    heada01-heada10 and data are not missing
1124  replace hemobsi = 1 if inlist(heada01,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1125  replace hemobsi = 1 if inlist(heada02,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1126  replace hemobsi = 1 if inlist(heada03,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1127  replace hemobsi = 1 if inlist(heada04,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1128  replace hemobsi = 1 if inlist(heada05,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1129  replace hemobsi = 1 if inlist(heada06,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1130  replace hemobsi = 1 if inlist(heada07,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1131  replace hemobsi = 1 if inlist(heada08,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1132  replace hemobsi = 1 if inlist(heada09,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1133  replace hemobsi = 1 if inlist(heada10,1,3,4,5,6,7,8,9,10,96) & hemobsi!=0
1134
1135  * [b18] Mobility: difficulty getting up from a chair after sitting for long periods
1136  replace hemobch = 2 if hemobch == 0
1137  * Assign the number 0 if the participant reported difficulties performing the third listed activity
1138  replace hemobch = 0 if hemobch == 1
1139  * Assign the number 1 if the participant reported no difficulties performing the third listed activity
1140  replace hemobch = 1 if hemobch == 2
1141  * Assign the number 0 if the participant reported difficulties performing the third listed activity
1142  replace hemobch = 0 if (heada01 == 3 | heada02 == 3 | heada03 == 3 | heada04 == 3 | heada05 == 3 |
    heada06 == 3 | heada07 == 3 | heada08 == 3 | heada09 == 3 | heada10 == 3) & wave==2
1143  * Assign the number 1 if the participant reported any answer other than the listed activity in
    heada01-heada10 and data are not missing
1144  replace hemobch = 1 if inlist(heada01,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1145  replace hemobch = 1 if inlist(heada02,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1146  replace hemobch = 1 if inlist(heada03,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1147  replace hemobch = 1 if inlist(heada04,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1148  replace hemobch = 1 if inlist(heada05,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1149  replace hemobch = 1 if inlist(heada06,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1150  replace hemobch = 1 if inlist(heada07,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1151  replace hemobch = 1 if inlist(heada08,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1152  replace hemobch = 1 if inlist(heada09,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1153  replace hemobch = 1 if inlist(heada10,1,2,4,5,6,7,8,9,10,96) & hemobch!=0
1154
1155  * [b19] Mobility: difficulty climbing several flights of stairs without resting
1156  replace hemobcs = 2 if hemobcs == 0
1157  * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
1158  replace hemobcs = 0 if hemobcs == 1
1159  * Assign the number 1 if the participant reported no difficulties performing the fourth listed
    activity
```

```
replace hemobcs = 1 if hemobcs == 2
* Assign the number 0 if the participant reported difficulties performing the fourth listed activity
replace hemobcs = 0 if (heada01 == 4 | heada02 == 4 | heada03 == 4 | heada04 == 4 | heada05 == 4 | heada06 == 4 | heada07 == 4 | heada08 == 4 | heada09 == 4 | heada10 == 4) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobcs = 1 if inlist(heada01,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada02,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada03,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada04,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada05,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada06,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada07,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada08,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada09,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
replace hemobcs = 1 if inlist(heada10,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0

* [b20] Mobility: difficulty climbing one flight of stairs without resting
replace hemobcl = 2 if hemobcl == 0
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace hemobcl = 0 if hemobcl == 1
* Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
replace hemobcl = 1 if hemobcl == 2
* Assign the number 0 if the participant reported difficulties performing the fifth listed activity
replace hemobcl = 0 if (heada01 == 5 | heada02 == 5 | heada03 == 5 | heada04 == 5 | heada05 == 5 | heada06 == 5 | heada07 == 5 | heada08 == 5 | heada09 == 5 | heada10 == 5) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobcl = 1 if inlist(heada01,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada02,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada03,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada04,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada05,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada06,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada07,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada08,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada09,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
replace hemobcl = 1 if inlist(heada10,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0

* [b21] Mobility: difficulty stooping, kneeling, or crouching
replace hemobst = 2 if hemobst == 0
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace hemobst = 0 if hemobst == 1
* Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
replace hemobst = 1 if hemobst == 2
* Assign the number 0 if the participant reported difficulties performing the sixth listed activity
replace hemobst = 0 if (heada01 == 6 | heada02 == 6 | heada03 == 6 | heada04 == 6 | heada05 == 6 | heada06 == 6 | heada07 == 6 | heada08 == 6 | heada09 == 6 | heada10 == 6) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobst = 1 if inlist(heada01,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada02,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada03,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada04,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada05,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada06,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada07,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada08,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada09,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
replace hemobst = 1 if inlist(heada10,1,2,3,4,5,7,8,9,10,96) & hemobst!=0

* [b22] Mobility: difficulty reaching or extending arms above shoulder level
replace hemobre = 2 if hemobre == 0
```

```
* Assign the number 0 if the participant reported difficulties performing the seventh listed activity
replace hemobre = 0 if hemobre == 1
* Assign the number 1 if the participant reported no difficulties performing the seventh listed activity
replace hemobre = 1 if hemobre == 2
* Assign the number 0 if the participant reported difficulties performing the seventh listed activity
replace hemobre = 0 if (heada01 == 7 | heada02 == 7 | heada03 == 7 | heada04 == 7 | heada05 == 7 | heada06 == 7 | heada07 == 7 | heada08 == 7 | heada09 == 7 | heada10 == 7) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobre = 1 if inlist(heada01,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada02,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada03,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada04,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada05,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada06,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada07,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada08,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada09,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
replace hemobre = 1 if inlist(heada10,1,2,3,4,5,6,8,9,10,96) & hemobre!=0

* [b23] Mobility: difficulty pulling or pushing large objects, like a living room chair
replace hemobpu = 2 if hemobpu == 0
* Assign the number 0 if the participant reported difficulties performing the eighth listed activity
replace hemobpu = 0 if hemobpu == 1
* Assign the number 1 if the participant reported no difficulties performing the eighth listed activity
replace hemobpu = 1 if hemobpu == 2
* Assign the number 0 if the participant reported difficulties performing the eighth listed activity
replace hemobpu = 0 if (heada01 == 8 | heada02 == 8 | heada03 == 8 | heada04 == 8 | heada05 == 8 | heada06 == 8 | heada07 == 8 | heada08 == 8 | heada09 == 8 | heada10 == 8) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobpu = 1 if inlist(heada01,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada02,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada03,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada04,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada05,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada06,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada07,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada08,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada09,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
replace hemobpu = 1 if inlist(heada10,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0

* [b24] Mobility: difficulty lifting or carrying weights over 10 pounds, like a heavy bag of groceries
replace hemobli = 2 if hemobli == 0
* Assign the number 0 if the participant reported difficulties performing the ninth listed activity
replace hemobli = 0 if hemobli == 1
* Assign the number 1 if the participant reported no difficulties performing the ninth listed activity
replace hemobli = 1 if hemobli == 2
* Assign the number 0 if the participant reported difficulties performing the ninth listed activity
replace hemobli = 0 if (heada01 == 9 | heada02 == 9 | heada03 == 9 | heada04 == 9 | heada05 == 9 | heada06 == 9 | heada07 == 9 | heada08 == 9 | heada09 == 9 | heada10 == 9) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in heada01-heada10 and data are not missing
replace hemobli = 1 if inlist(heada01,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada02,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada03,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada04,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada05,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada06,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada07,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada08,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
```

```
replace hemobli = 1 if inlist(heada09,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
replace hemobli = 1 if inlist(heada10,1,2,3,4,5,6,7,8,10,96) & hemobli!=0

* [b25] Mobility: difficulty picking up a 5p coin from a table
replace hemobpi = 2 if hemobpi == 0
* Assign the number 0 if the participant reported difficulties performing the tenth listed activity
replace hemobpi = 0 if hemobpi == 1
* Assign the number 1 if the participant reported no difficulties performing the tenth listed activity
replace hemobpi = 1 if hemobpi == 2
* Assign the number 0 if the participant reported difficulties performing the tenth listed activity
replace hemobpi = 0 if (heada01 == 10 | heada02 == 10 | heada03 == 10 | heada04 == 10 | heada05 == 10
 | heada06 == 10 | heada07 == 10 | heada08 == 10 | heada09 == 10 | heada10 == 10) & wave==2
* Assign the number 1 if the participant reported any answer other than the listed activity in
heada01-heada10 and data are not missing
replace hemobpi = 1 if inlist(heada01,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada02,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada03,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada04,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada05,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada06,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada07,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada08,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada09,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
replace hemobpi = 1 if inlist(heada10,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0

* Overwrite dataset, by replacing the previously saved file
save datavariables.dta, replace

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* Socio-economic covariate - Quintiles of BU total (non-pension) wealth
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace totwq5_bu_s = . if totwq5_bu_s<0

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b41] Whether has self-reported limiting long-standing illness (reversed)
* Generate a new variable and assign the number 0 for participants with a limiting long-standing
illness
gen limiting = 0 if helim==1
* Assign the number 1 for participants with no long-standing illness or a long-standing illness that
is not limiting
replace limiting = 1 if heill == 2 | helim == 2

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b9-b15] (+ 1 omitted) Organisational memberships
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
* [b9] Political party, trade union or environmental group
replace scorg01 = . if scorg01<0
* [b10] Tenants or resident group or neighbourhood watch
replace scorg02 = . if scorg02<0
* [b11] Member of a church or other religious group
replace scorg03 = . if scorg03<0
* [b12] Member of a charitable association
replace scorg04 = . if scorg04<0
* [b13] An education, arts or music group or evening class
replace scorg05 = . if scorg05<0
* (1 omitted) Member of a social club
replace scorg06 = . if scorg06<0
* [b14] Member of a sports club, gym, or exercise class
replace scorg07 = . if scorg07<0
* [b15] Member of any other organisations, clubs, or societies
replace scorg08 = . if scorg08<0

* Save dataset with a new name
```

```
save datavariables01.dta

* [b42-b43] (+ 1 omitted) Short Physical Performance Battery
* WAVE 2, 4, 6
* (1 omitted) Standing balance
tab mmbcsc
tab mmsssc
tab mmssre
tab mmsssc if mmssre==3
tab mmbcsc if mmsssc<0
tab mmsssc if mmssre<0
sum mmssti if mmssti!=-1
tab mmssre if mmssti!=-1
sum mmssna if mmssna!=-1
tab mmssna if mmssna!=-1 & wave==2
tab mmssna if mmssna!=-1 & inlist(wave,4,6)
tab mmsssc if mmssna!=-1

tab mmstsc
tab mmstsc if mmssna!=-1
tab mmssre if mmstsc==-1
tab mmstre
tab mmstsc if mmstre==3
sum mmstti if mmstti!=-1
tab mmstre if mmstti!=-1
sum mmstna if mmstna!=-1
tab mmstna if mmstna!=-1 & wave==2
tab mmstna if mmstna!=-1 & inlist(wave,4,6)
tab mmstsc if mmstna!=-1

tab mmftsc
tab mmftsc if mmstna!=-1
tab mmftsc if mmstti!=-1
tab mmftsc if mmssna!=-1 | mmssti!=-1
tab mmftsc if mmsssc<0
tab mmftre2
tab mmftsc if mmftre2==5
sum mmftti if mmftti!=-1
sum mmftti if inlist(mmftre2,2,4)
tab mmftre2 if mmftti!=-1
tab mmftre2 if inlist(mmftre2,2,4)
sum mmftti if mmftti >=3 & mmftti < 10
sum mmftti if mmftti < 3 & mmftti!=-1
tab mmftre2 if mmftti >=10 & mmftti!=.
sum mmftti if mmftti >=10 & mmftti!=.
sum mmftna if mmftna!=-1
tab mmftna if mmftna!=-1 & wave==2
tab mmftna if mmftna!=-1 & inlist(wave,4,6)
tab mmftsc if mmftna!=-1
tab mmftre2 if wave==6
tab mmftre

* Side-by-side stand
* Generate a new variable and assign the number 0 if the participant held for less than 10 seconds
or did not attempt the stand
gen sidebyside = 0 if inlist(mmssre,2,3)
* Assign the number 1 if the participant held for 10 seconds
replace sidebyside = 1 if mmssre==1

* Semi-tandem stand
* Generate a new variable and assign the number 0 if the participant scored 0 points in the
side-by-side stand
gen semitandem = 0 if sidebyside==0
```

```
* Assign the number 0 if the participant held for less than 10 seconds or did not attempt the stand
replace semitandem = 0 if inlist(mmstre,2,3)
* Assign the number 1 if the participant held for 10 seconds
replace semitandem = 1 if mmstre==1

* Full tandem stand
* Generate a new variable and assign the number 0 if the participant did not attempt the stand or scored 0 points in the semi-tandem stand
gen tandem = 0 if (mmftre2 == 5 |semitandem == 0)
* Assign the number 2 if the participant held for at least 10 seconds, irrespective of age
replace tandem = 2 if (mmftre2 == 1 |mmftre2 == 3)
* Assign the number 1 if the participant held for at least 3 seconds but less than 10 seconds
replace tandem = 1 if mmftti >=3 & mmftti < 10
* Assign the number 0 if the participant held for less than 3 seconds and data are not missing
replace tandem = 0 if mmftti < 3 & mmftti!=-1
* Assign the number 2 if the participant held for at least 10 seconds (but less than 30 seconds) and was aged less than 70 years
replace tandem = 2 if mmftti >=10 & mmftti!=. & mmftre2==4

* Generate a new variable equal to the sum of the points scored on the side-by-side, semi-tandem, and full tandem stands (three items) to create a total balance score (range 0-4)
gen balance = sidebyside + semitandem + tandem

gen balance2 = 0 if inlist(mmssre,2,3)
replace balance2 = 1 if mmssre==1 & inlist(mmstre,2,3)
replace balance2 = 2 if mmstre==1 & tandem==0
replace balance2 = 3 if mmstre==1 & tandem==1
replace balance2 = 4 if mmstre==1 & tandem==2

* WAVE 2, 4, 6
* [b42] Repeated chair stands
tab mmcrav
* Generate a new variable and assign a missing value if there was no suitable chair available or data are missing
gen repcstest = . if mmcrav==2 | mmcrav<0
tab mmcrsc
tab mmcrsc if mmcrav==2 | mmcrav<0
tab mmcrre
tab mmcrre if mmcrav==2 | mmcrav<0 | inlist(mmcrsc,-8,2)
* Assign the number 0 if the participant did not feel it was safe to attempt the single chair rise or the response was coded "Don't know"
replace repcstest = 0 if inlist(mmcrsc,-8,2)
* Assign the number 0 if the participant used their arms to stand in the single chair rise or did not complete the test
replace repcstest = 0 if inlist(mmcrre,2,3)
tab mmcrav if mmcrre==-1
tab mmcrsc if mmcrre==-1
sum mmcrna if mmcrna!=-1
tab mmcrna if mmcrna!=-1 & wave==2
tab mmcrna if mmcrna!=-1 & inlist(wave,4,6)
tab mmrrsc
tab mmcrre if mmrrsc==-1
tab mmrrre
tab mmrrsc if mmrrre==-1
tab mmrrre if inlist(mmrrsc,2,-1)
* Assign the number 0 if the participant did not feel it was safe to attempt multiple chair rises (and subsequently did not perform the multiple chair rise test)
replace repcstest = 0 if mmrrsc==2 & mmrrre==-1
* Assign the number 0 if the participant completed less than five sit-to-stands
replace repcstest = 0 if inlist(mmrrre,0,1,2,3,4)
tab mmrrfti if mmrrfti<0
tab mmrrre if mmrrfti<0
sum mmrrfti if mmrrre>=5
```

```
sum mmrrfti if mmrrfti>=0
tab mmrrfti if mmrrfti<0 & wave==2
tab mmrrfti if mmrrfti<0 & wave==4
tab mmrrfti if mmrrfti<0 & wave==6
* Assign a missing value if the participant completed five or more sit-to-stands but their time to complete five rises was coded as "Don't know" or the test was not timed correctly
replace repcstest = . if inlist(mmrrfti,-8,-3) & mmrrre>=5
tab mmrrfti if mmrrfti>=0 & mmrrfti<4
* Assign a missing value if the participant completed five or more sit-to-stands but their time to complete five rises was equal to 0 or 1 seconds (i.e., outlier)
replace repcstest = . if inlist(mmrrfti,0,1) & inlist(mmrrre,5,6,7,8,9,10)
sum mmrrfti if mmrrfti<=11.19 & mmrrfti>=0
sum mmrrfti if mmrrfti<=11.19 & mmrrfti>1
* Assign the number 4 if the participant completed five sit-to-stands in less than or equal to 11.19 seconds and it took them more than 1 second
replace repcstest = 4 if mmrrfti <= 11.19 & mmrrfti > 1 & inlist(mmrrre,5,6,7,8,9,10)
sum mmrrfti if mmrrfti>=16.7 & mmrrfti<=60
* Assign the number 1 if the participant completed five sit-to-stands in 16.7 seconds or more but less than or equal to 60 seconds
replace repcstest = 1 if mmrrfti >= 16.7 & mmrrfti <= 60 & inlist(mmrrre,5,6,7,8,9,10)
sum mmrrfti if mmrrfti>=13.7 & mmrrfti<16.7
* Assign the number 2 if the participant completed five sit-to-stands in 13.7 seconds or more but less than 16.7 seconds
replace repcstest = 2 if mmrrfti >= 13.7 & mmrrfti < 16.7 & inlist(mmrrre,5,6,7,8,9,10)
sum mmrrfti if mmrrfti>=11.2 & mmrrfti<13.7
* Assign the number 3 if the participant completed five sit-to-stands in 11.2 seconds or more but less than 13.7 seconds
replace repcstest = 3 if mmrrfti >= 11.2 & mmrrfti < 13.7 & inlist(mmrrre,5,6,7,8,9,10)
sum mmrrfti if mmrrfti > 60
* Assign the number 0 if the participant completed five sit-to-stands in more than 60 seconds and data are not missing
replace repcstest = 0 if mmrrfti > 60 & mmrrfti <= 64 & inlist(mmrrre,5,6,7,8,9,10)

tab mmrrre if inlist(mmrroc,1,3)
tab mmrrre if mmrroc==2
tab mmrrre if mmrroc==4
tab mmrrre if inlist(mmrroc,1,2,3,4)

sum mmrrna if mmrrna!=-1
tab mmrrna if mmrrna!=-1 & wave==2
tab mmrrna if mmrrna!=-1 & inlist(wave,4,6)
tab mmrrre if mmrrna!=-1 & inlist(mmrroc,3,4)
tab mmrrre if mmrrna!=-1 & inlist(mmrroc,1,2)
tab mmrrsc if mmrrna!=-1
tab mmrrna if mmrrre==5 & inlist(mmrroc,1,2)
sum mmrrfti if mmrrre==5 & inlist(mmrroc,1,2) & mmrrna!=-1

* Save dataset with a new name
save datavariables02.dta

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b43] Gait speed
tab mmschs if indager >=60
tab mmalone if indager >=60
tab mmschs if mmalone==-1 & indager >=60
tab mmschs if mmalone==-9 & indager >=60
tab mmschs if mmalone==-8 & indager >=60
tab mmschs if mmalone==-2 & indager >=60
tab mmschs if mmalone==1 & indager >=60
tab mmschs if mmalone==2 & indager >=60
tab mmschs if mmalone==3 & indager >=60
tab mmhss if indager >=60
tab mmalone if mmhss==-1 & indager >=60
```

```
tab mmalone if mmhss==-9 & indager >=60
tab mmalone if mmhss==-8 & indager >=60
tab mmalone if mmhss==-2 & indager >=60
tab mmalone if mmhss==1 & indager >=60
tab mmalone if mmhss==2 & indager >=60
tab mmalone if mmhss==3 & indager >=60
tab mmalone if mmhss==4 & indager >=60
tab mmwill if indager >=60
tab mmhss if mmwill==-9 & indager >=60
tab mmhss if mmwill==-8 & indager >=60
tab mmhss if mmwill==-2 & indager >=60
tab mmhss if mmwill==-1 & indager >=60
tab mmhss if mmwill==1 & indager >=60
tab mmhss if mmwill==2 & indager >=60
tab mmsaf if indager >=60
tab mmwill if mmsaf==1 & indager >=60
tab mmwill if mmsaf==2 & indager >=60
tab mmavsp if indager >=60
tab mmsaf if mmavsp==-9 & indager >=60
tab mmsaf if mmavsp==-8 & indager >=60
tab mmsaf if mmavsp==-2 & indager >=60
tab mmsaf if mmavsp==-1 & indager >=60
tab mmsaf if mmavsp==1 & indager >=60
tab mmsaf if mmavsp==2 & indager >=60
tab mmwala if indager >=60
tab mmavsp if inlist(mmwala,1,2) & indager >=60

tab mmtrya if indager >=60
tab mmwala if inlist(mmtrya,-9,1,2,3) & indager >=60
tab mmwala if mmtrya==4 & indager >=60
tab mmwala if mmtrya==-8 & indager >=60
tab mmwala if mmtrya==-2 & indager >=60
tab mmwlka if mmwlka<0 & indager >=60
sum mmwlka if mmwlka>=0 & indager >=60
tab mmtrya if mmwlka==-1 & indager >=60
tab mmtrya if mmwlka==-2 & indager >=60
tab mmtrya if mmwlka==-8 & indager >=60
tab mmtrya if mmwlka==-9 & indager >=60
tab mmtrya if mmwlka>=0 & indager >=60

tab mmtryb if indager >=60
tab mmtrya if inlist(mmtryb,-8,1,2,3,4) & indager >=60
tab mmwlka if inlist(mmtryb,-9,-8,-2,-1) & indager >=60
tab mmwlkb if mmwlkb<0 & indager >=60
sum mmwlkb if mmwlkb>=0 & indager >=60
tab mmtryb if mmwlkb==-1 & indager >=60
tab mmtryb if mmwlkb==-2 & indager >=60
tab mmtryb if mmwlkb==-8 & indager >=60
tab mmtryb if mmwlkb==-9 & indager >=60
tab mmtryb if mmwlkb>=0 & indager >=60

tab mmwlka if mmwlka>=0 & mmwlka<2 & indager >=60
tab mmwlkb if mmwlkb>=0 & mmwlkb<2 & indager >=60

sum idauniq if ((mmwlka>=0 & mmwlka!=.) | (mmwlkb>=0 & mmwlkb!=.)) & indager >=60
sum idauniq if mmwlka>=0 & mmwlkb>=0 & mmwlka!=. & mmwlkb!=. & indager >=60

sum idauniq if mmwlka<0 & indager >=60
keep if mmwlka<0 & indager >=60
tab mmschs if indager >=60
tab mmalone if indager >=60
tab mmschs if mmalone==-1 & indager >=60
tab mmschs if mmalone==3 & indager >=60
```

```
tab mmschs if mmalone==-9 & indager >=60
tab mmschs if mmalone==-8 & indager >=60
tab mmschs if mmalone==-2 & indager >=60
tab mmhss if indager >=60
tab mmalone if mmhss==-1 & indager >=60
tab mmwill if indager >=60
tab mmhss if mmwill==-1 & indager >=60
tab mmsaf if indager >=60
tab mmwill if mmsaf==-1 & indager >=60
tab mmavsp if indager >=60
tab mmsaf if mmavsp==-1 & indager >=60
tab mmwala if indager >=60
tab mmavsp if mmwala==-1 & indager >=60
tab mmtrya if indager >=60
tab mmwala if mmtrya==-1 & indager >=60
tab mmwlka if indager >=60
tab mmtrya if mmwlka==-1 & indager >=60
tab mmtrya if mmwlka<0 & mmwlka!=-1 & indager >=60

clear
* Use full dataset
use datavariables02.dta

* Generate a new variable duplicating the "time taken for first walk" variable if data are not missing
gen walk1 = mmwlka if mmwlka>=0 & indager >=60
* Generate a new variable duplicating the "time taken for second walk" variable if data are not
missing
gen walk2 = mmwlkb if mmwlkb>=0 & indager >=60
* Generate a new variable equal to the fastest time of the two walks (or the only time available if
only one attempt was performed or recorded)
egen gaittime = rowmin(walk1 walk2) if ((mmwlka>=0 & mmwlka!=.) | (mmwlkb>=0 & mmwlkb!=.)) & indager
>=60
sum gaittime
sum gaittime if (mmwlka>=0 & mmwlka!=.) & (mmwlkb<0|mmwlkb==.)
sum mmwlka if (mmwlka>=0 & mmwlka!=.) & (mmwlkb<0|mmwlkb==.)

* Generate a new variable
gen gaittest = .
* Assign the number 1 if the participant completed the gait test in more than or equal to 5.7 seconds
replace gaittest = 1 if gaittime >= 5.7 & gaittime!=.
* Assign the number 2 if the participant completed the gait test in more than or equal to 4.1
seconds and less than 5.7 seconds
replace gaittest = 2 if gaittime >= 4.1 & gaittime < 5.7
* Assign the number 3 if the participant completed the gait test in more than or equal to 3.2
seconds and less than 4.1 seconds
replace gaittest = 3 if gaittime >= 3.2 & gaittime < 4.1
* Assign the number 4 if the participant completed the gait test in less than 3.2 seconds
replace gaittest = 4 if gaittime < 3.2
* Assign the number 0 if a) the participant was not able to walk alone (with aid); b) a health
condition (i.e., recent surgery, injury, other health condition) prevented the participant from
walking; c) the interviewer felt it was not safe to continue the test; d) the respondent did not
feel the walk would be safe; or e) the participant attempted the walk but was unable to complete it
or was stopped by the interviewer because of safety reasons
replace gaittest = 0 if (mmalone==3 | inlist(mmhss,2,3,4) | mmsaf==2  | mmwala==2 | inlist(mmtrya,2,3
)) & indager >=60

* Save dataset with a new name
save datavariables03.dta

* WAVE 2, 4, 5, 6, 7, 8, 9
* [b44] Self-reported general health (reversed)
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace hehelf = . if hehelf<0
```

```
* Reverse the self-rated health variable
revrs hehelf

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b1-b6] (+ 2 omitted) Depressive symptoms
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
* [b1] Whether respondent has felt depressed much of the time during the past week
replace psceda = . if psceda<0
* (1 omitted) Whether respondent felt everything they did during the past week was an effort
replace pscedb = . if pscedb<0
* [b2] Whether respondent felt their sleep was restless during the past week
replace pscedc = . if pscedc<0
* [b3] Whether respondent was happy much of the time during the past week (reversed)
replace pscedd = . if pscedd<0
* [b4] Whether respondent felt lonely much of the time during the past week
replace pscede = . if pscede<0
* [b5] Whether respondent enjoyed life much of the time during the past week (reversed)
replace pscedf = . if pscedf<0
* [b6] Whether respondent felt sad much of the time during the past week
replace pscedg = . if pscedg<0
* (1 omitted) Whether respondent could not get going much of the time during the past week
replace pscedh = . if pscedh<0

* Recode to the number 0 if participant answered "Yes"
replace psceda = 0 if psceda == 1
* Recode to the number 1 if participant answered "No"
replace psceda = 1 if psceda == 2

* Recode to the number 0 if participant answered "Yes"
replace pscedb = 0 if pscedb == 1
* Recode to the number 1 if participant answered "No"
replace pscedb = 1 if pscedb == 2

* Recode to the number 0 if participant answered "Yes"
replace pscedc = 0 if pscedc == 1
* Recode to the number 1 if participant answered "No"
replace pscedc = 1 if pscedc == 2

* Recode to the number 0 if participant answered "Yes"
replace pscede = 0 if pscede == 1
* Recode to the number 1 if participant answered "No"
replace pscede = 1 if pscede == 2

* Recode to the number 0 if participant answered "Yes"
replace pscedg = 0 if pscedg == 1
* Recode to the number 1 if participant answered "No"
replace pscedg = 1 if pscedg == 2

* Recode to the number 0 if participant answered "Yes"
replace pscedh = 0 if pscedh == 1
* Recode to the number 1 if participant answered "No"
replace pscedh = 1 if pscedh == 2

* Recode to the number 0 if participant answered "No"
replace pscedd = 0 if pscedd == 2
* Recode to the number 0 if participant answered "No"
replace pscedf = 0 if pscedf == 2

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* (3 omitted) Loneliness (reversed)
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
* (1 omitted) How often respondent feels they lack companionship
replace scfeela = . if scfeela<0
```

```
* (1 omitted) How often respondent feels left out
replace scfeelb = . if scfeelb<0
* (1 omitted) How often respondent feels isolated from others
replace scfeelc = . if scfeelc<0

* Reverse the negatively framed variables
revrs scfeela
revrs scfeelb
revrs scfeelc

* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* Outcome variable - Quality-of-life
* Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace scqola = . if scqola<0
replace scqolb = . if scqolb<0
replace scqolc = . if scqolc<0
replace scqold = . if scqold<0
replace scqole = . if scqole<0
replace scqolf = . if scqolf<0
replace scqolg = . if scqolg<0
replace scqolh = . if scqolh<0
replace scqoli = . if scqoli<0
replace scqolj = . if scqolj<0
replace scqolk = . if scqolk<0
replace scqoll = . if scqoll<0
replace scqolm = . if scqolm<0
replace scqoln = . if scqoln<0
replace scqolo = . if scqolo<0
replace scqolp = . if scqolp<0
replace scqolq = . if scqolq<0
replace scqolr = . if scqolr<0
replace scqols = . if scqols<0

* Recode each item into a 0-3 scale (13 of the 19 items were reversed)
replace scqola = 0 if scqola == 1
replace scqola = 1 if scqola == 2
replace scqola = 2 if scqola == 3
replace scqola = 3 if scqola == 4
replace scqolb = 0 if scqolb == 1
replace scqolb = 1 if scqolb == 2
replace scqolb = 2 if scqolb == 3
replace scqolb = 3 if scqolb == 4
revrs scqolc
replace revscqolc = 0 if revscqolc == 1
replace revscqolc = 1 if revscqolc == 2
replace revscqolc = 2 if revscqolc == 3
replace revscqolc = 3 if revscqolc == 4
replace scqold = 0 if scqold == 1
replace scqold = 1 if scqold == 2
replace scqold = 2 if scqold == 3
replace scqold = 3 if scqold == 4
revrs scqole
replace revscqole = 0 if revscqole == 1
replace revscqole = 1 if revscqole == 2
replace revscqole = 2 if revscqole == 3
replace revscqole = 3 if revscqole == 4
replace scqolf = 0 if scqolf == 1
replace scqolf = 1 if scqolf == 2
replace scqolf = 2 if scqolf == 3
replace scqolf = 3 if scqolf == 4
revrs scqolg
replace revscqolg = 0 if revscqolg == 1
replace revscqolg = 1 if revscqolg == 2
```

```
1746   replace revscqolg = 2 if revscqolg == 3
1747   replace revscqolg = 3 if revscqolg == 4
1748   replace scqolh = 0 if scqolh == 1
1749   replace scqolh = 1 if scqolh == 2
1750   replace scqolh = 2 if scqolh == 3
1751   replace scqolh = 3 if scqolh == 4
1752   replace scqoli = 0 if scqoli == 1
1753   replace scqoli = 1 if scqoli == 2
1754   replace scqoli = 2 if scqoli == 3
1755   replace scqoli = 3 if scqoli == 4
1756   revrs scqolj
1757   replace revscqolj = 0 if revscqolj == 1
1758   replace revscqolj = 1 if revscqolj == 2
1759   replace revscqolj = 2 if revscqolj == 3
1760   replace revscqolj = 3 if revscqolj == 4
1761   revrs scqolk
1762   replace revscqolk = 0 if revscqolk == 1
1763   replace revscqolk = 1 if revscqolk == 2
1764   replace revscqolk = 2 if revscqolk == 3
1765   replace revscqolk = 3 if revscqolk == 4
1766   revrs scqoll
1767   replace revscqoll = 0 if revscqoll == 1
1768   replace revscqoll = 1 if revscqoll == 2
1769   replace revscqoll = 2 if revscqoll == 3
1770   replace revscqoll = 3 if revscqoll == 4
1771   revrs scqolm
1772   replace revscqolm = 0 if revscqolm == 1
1773   replace revscqolm = 1 if revscqolm == 2
1774   replace revscqolm = 2 if revscqolm == 3
1775   replace revscqolm = 3 if revscqolm == 4
1776   revrs scqoln
1777   replace revscqoln = 0 if revscqoln == 1
1778   replace revscqoln = 1 if revscqoln == 2
1779   replace revscqoln = 2 if revscqoln == 3
1780   replace revscqoln = 3 if revscqoln == 4
1781   revrs scqolo
1782   replace revscqolo = 0 if revscqolo == 1
1783   replace revscqolo = 1 if revscqolo == 2
1784   replace revscqolo = 2 if revscqolo == 3
1785   replace revscqolo = 3 if revscqolo == 4
1786   revrs scqolp
1787   replace revscqolp = 0 if revscqolp == 1
1788   replace revscqolp = 1 if revscqolp == 2
1789   replace revscqolp = 2 if revscqolp == 3
1790   replace revscqolp = 3 if revscqolp == 4
1791   revrs scqolq
1792   replace revscqolq = 0 if revscqolq == 1
1793   replace revscqolq = 1 if revscqolq == 2
1794   replace revscqolq = 2 if revscqolq == 3
1795   replace revscqolq = 3 if revscqolq == 4
1796   revrs scqolr
1797   replace revscqolr = 0 if revscqolr == 1
1798   replace revscqolr = 1 if revscqolr == 2
1799   replace revscqolr = 2 if revscqolr == 3
1800   replace revscqolr = 3 if revscqolr == 4
1801   revrs scqols
1802   replace revscqols = 0 if revscqols == 1
1803   replace revscqols = 1 if revscqols == 2
1804   replace revscqols = 2 if revscqols == 3
1805   replace revscqols = 3 if revscqols == 4
1806
1807   * Generate a new variable equal to the sum of the 19 items (range 0-57)
1808   gen QoL = scqola + scqolb + revscqolc + scqold + revscqole + scqolf + revscqolg + scqolh + scqoli +
```

```
    revscqolj + revscqolk + revscqoll + revscqolm + revscqoln + revscqolo + revscqolp + revscqolq +
    revscqolr + revscqols
1809
1810    * Overwrite dataset, by replacing the previously saved file
1811    save datavariables03.dta, replace
1812
1813    * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1814    * Independent variable - Alcohol consumption
1815    * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1816    replace scako = . if scako<0
1817    * Assign the number 0 if the participant reported having an alcoholic drink once a month or less
    during the last 12 months
1818    replace scako = 0 if inlist(scako,5,6,7,8)
1819    * Assign the number 2 if the participant reported having an alcoholic drink a) almost every day; or
    b) five or six days a week
1820    replace scako = 2 if scako==1
1821    * Assign the number 1 if the participant reported having an alcoholic drink a) three or four days a
    week; or b) once or twice a week
1822    replace scako = 1 if inlist(scako,3,4)
1823
1824    * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1825    * Socio-economic covariate - Education
1826    * Generate a new variable
1827    gen education = .
1828    * Assign the number 0 if the participant does not have any formal qualifications
1829    replace education = 0 if edqual==7
1830    * Assign the number 1 if the participant has A level equivalent, O level equivalent, or other grade
    equivalent
1831    replace education = 1 if inlist(edqual,3,4,5)
1832    * Assign the number 2 if the participant has completed some higher education (below degree), or has
    a degree or equivalent
1833    replace education = 2 if inlist(edqual,1,2)
1834
1835    * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1836    * Demographic covariate - Ethnicity
1837    * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1838    replace fqethnr = . if fqethnr<0
1839    * Assign the number 0 if the participant is White
1840    replace fqethnr = 0 if fqethnr==1
1841    * Assign the number 1 if the participant is Non-White
1842    replace fqethnr = 1 if fqethnr==2
1843
1844    * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1845    * Demographic covariate - Biological sex
1846    * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1847    replace disex = . if disex<0
1848    * Assign the number 0 if the participant is male
1849    replace disex = 0 if disex==1
1850    * Assign the number 1 if the participant is female
1851    replace disex = 1 if disex==2
1852
1853    * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1854    * Independent variable - Physical activity
1855    * Generate a new variable
1856    gen activity2 = .
1857    * Assign the number 3 if the participant partakes in vigorous activity more than once a week or once
    a week
1858    replace activity2 = 3 if heacta==1 | heacta==2
1859    * Assign the number 2 if the participant partakes in moderate activity more than once a week or once
    a week, and takes part in vigorous activity less than once a week
1860    replace activity2 = 2 if (heactb==1 | heactb==2) & inlist(heacta,3,4)
1861    * Assign the number 1 if the participant partakes in mild activity more than once a week or once a
    week, and takes part in moderate and vigorous activities less than once a week
```

```
1862  replace activity2 = 1 if (heactc==1 | heactc==2) & inlist(heacta,3,4) & inlist(heactb,3,4)
1863  * Assign the number 0 if the participant does not take part in activity of any intensity once a week
      or more
1864  replace activity2 = 0 if inlist(heacta,3,4) & inlist(heactb,3,4) & inlist(heactc,3,4)
1865  * Replace the variable as missing for participants with missing cases on all three variables
1866  replace activity2 = . if inlist(heacta,.) & inlist(heactb,.) & inlist(heactc,.)
1867
1868  * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1869  * Independent variable - Smoking status
1870  * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
      wave) variable at Wave 2
1871  gen heske2 = heske if wave==2
1872  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1873  tsset idauniq wave
1874  * Generate a completely balanced dataset (i.e., all participants have a row for each wave from 2 to 9)
1875  tsfill, full
1876  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves) by participant ID
1877  bysort idauniq: carryforward heske2, replace
1878
1879  * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
      wave) variable at Wave 3
1880  gen heske3 = heske if wave==3
1881  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1882  tsset idauniq wave
1883  * Generate a completely balanced dataset
1884  tsfill, full
1885  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the
      follow-up waves) by participant ID
1886  bysort idauniq: carryforward heske3, replace
1887  * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
      values of the time variable "wave"
1888  gsort idauniq - wave
1889  * Carryforward (in a backward way) observations with respect to the time variable "wave" by
      participant ID
1890  bysort idauniq: carryforward heske3, replace
1891
1892  * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
      wave) variable at Wave 4
1893  gen heske4 = heske if wave==4
1894  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1895  tsset idauniq wave
1896  * Generate a completely balanced dataset
1897  tsfill, full
1898  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the
      follow-up waves) by participant ID
1899  bysort idauniq: carryforward heske4, replace
1900  * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
      values of the time variable "wave"
1901  gsort idauniq - wave
1902  * Carryforward (in a backward way) observations with respect to the time variable "wave" by
      participant ID
1903  bysort idauniq: carryforward heske4, replace
1904
1905  * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
      wave) variable at Wave 5
1906  gen heske5 = heske if wave==5
1907  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1908  tsset idauniq wave
1909  * Generate a completely balanced dataset
1910  tsfill, full
1911  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 5 to the
      follow-up waves) by participant ID
```

```
bysort idauniq: carryforward heske5, replace
* Sort data in memory by ascending values of the participant ID variable "idauniq" and descending values of the time variable "wave"
gsort idauniq - wave
* Carryforward (in a backward way) observations with respect to the time variable "wave" by participant ID
bysort idauniq: carryforward heske5, replace

* Generate a new variable duplicating the heske (reason disputed reported smoking from previous wave) variable at Wave 6
gen heske6 = heske if wave==6
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Generate a completely balanced dataset
tsfill, full
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves) by participant ID
bysort idauniq: carryforward heske6, replace
* Sort data in memory by ascending values of the participant ID variable "idauniq" and descending values of the time variable "wave"
gsort idauniq - wave
* Carryforward (in a backward way) observations with respect to the time variable "wave" by participant ID
bysort idauniq: carryforward heske6, replace

* Generate a new variable duplicating the heske (reason disputed reported smoking from previous wave) variable at Wave 7
gen heske7 = heske if wave==7
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Generate a completely balanced dataset
tsfill, full
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves) by participant ID
bysort idauniq: carryforward heske7, replace
* Sort data in memory by ascending values of the participant ID variable "idauniq" and descending values of the time variable "wave"
gsort idauniq - wave
* Carryforward (in a backward way) observations with respect to the time variable "wave" by participant ID
bysort idauniq: carryforward heske7, replace

* Generate a new variable duplicating the heske (reason disputed reported smoking from previous wave) variable at Wave 8
gen heske8 = heske if wave==8
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Generate a completely balanced dataset
tsfill, full
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 8 to the follow-up wave) by participant ID
bysort idauniq: carryforward heske8, replace
* Sort data in memory by ascending values of the participant ID variable "idauniq" and descending values of the time variable "wave"
gsort idauniq - wave
* Carryforward (in a backward way) observations with respect to the time variable "wave" by participant ID
bysort idauniq: carryforward heske8, replace

* Generate a new variable duplicating the heske (reason disputed reported smoking from previous wave) variable at Wave 9
gen heske9 = heske if wave==9
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
```

```
tsset idauniq wave
* Generate a completely balanced dataset
tsfill, full
* Sort data in memory by ascending values of the participant ID variable "idauniq" and descending values of the time variable "wave"
gsort idauniq - wave
* Carryforward (in a backward way) observations with respect to the time variable "wave" by participant ID
bysort idauniq: carryforward heske9, replace

* Generate a new variable duplicating the bhesmkc (whether reported ever smoked cigarettes at Wave 1) variable at Wave 2
gen bhesmkc = bhesmk
* Assign the number 1 if the participant reported no longer smoking cigarettes by last interview (Wave 1)
replace bhesmkc = 1 if heske==2 & wave==2
* Assign the number 2 if the participant reported never having smoked cigarettes
replace bhesmkc = 2 if heske==1 & wave==2

* Generate a new variable and assign the number 0 if the participant reported never having smoked cigarettes
gen smoking = 0 if hesmk==2
* Assign the number 0 if the participant reported never having smoked cigarettes at Wave 1 and reported that they do not smoke cigarettes at all nowadays
replace smoking = 0 if bhesmkc==2 & heska==2
* Assign the number 1 if the participant reported having ever smoked cigarettes but reported that they do not smoke cigarettes at all nowadays
replace smoking = 1 if (hesmk==1 | bhesmkc==1) & heska==2
* Assign the number 2 if the participant reported smoking nowadays
replace smoking = 2 if heska==1

* Assign the number 1 if the participant reported that they stopped smoking between Wave 1 and Wave 2 and reported that they do not smoke cigarettes at all nowadays at Wave 2
replace smoking = 1 if heske==3 & heska==2 & wave==2
tab heskd
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 2
replace smoking = 0 if heske3==1 & heska==2 & wave==2
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 2 and reported that they do not smoke cigarettes at all nowadays at Wave 2
replace smoking = 1 if heske3==2 & heska==2 & wave==2

* Assign the number 1 if the participant reported that they stopped smoking between Wave 2 and Wave 3 and reported that they do not smoke cigarettes at all nowadays at Wave 3
replace smoking = 1 if heske==3 & heska==2 & wave==3
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 3
replace smoking = 0 if heske4==1 & heska==2 & wave==3
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 3 and reported that they do not smoke cigarettes at all nowadays at Wave 3
replace smoking = 1 if heske4==2 & heska==2 & wave==3

* Assign the number 1 if the participant reported that they stopped smoking between Wave 3 and Wave 4 and reported that they do not smoke cigarettes at all nowadays at Wave 4
replace smoking = 1 if heske==3 & heska==2 & wave==4
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 4
replace smoking = 0 if heske5==1 & heska==2 & wave==4
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 4 and reported that they do not smoke cigarettes at all nowadays at Wave 4
replace smoking = 1 if heske5==2 & heska==2 & wave==4

tab heske6
```

```
tab heske5
tab heske7
* Assign the number 1 if the participant reported that they stopped smoking between Wave 4 and Wave 5 and reported that they do not smoke cigarettes at all nowadays at Wave 5
replace smoking = 1 if heske==3 & heska==2 & wave==5
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 5
replace smoking = 0 if heske6==1 & heska==2 & wave==5
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 5 and reported that they do not smoke cigarettes at all nowadays at Wave 5
replace smoking = 1 if heske6==2 & heska==2 & wave==5

* Assign the number 1 if the participant reported that they stopped smoking between Wave 5 and Wave 6 and reported that they do not smoke cigarettes at all nowadays at Wave 6
replace smoking = 1 if heske==3 & heska==2 & wave==6
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 6
replace smoking = 0 if heske7==1 & heska==2 & wave==6
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 6 and reported that they do not smoke cigarettes at all nowadays at Wave 6
replace smoking = 1 if heske7==2 & heska==2 & wave==6

* Assign the number 1 if the participant reported that they stopped smoking between Wave 6 and Wave 7 and reported that they do not smoke cigarettes at all nowadays at Wave 7
replace smoking = 1 if heske==3 & heska==2 & wave==7
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 7
replace smoking = 0 if heske8==1 & heska==2 & wave==7
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 7 and reported that they do not smoke cigarettes at all nowadays at Wave 7
replace smoking = 1 if heske8==2 & heska==2 & wave==7

* Assign the number 1 if the participant reported that they stopped smoking between Wave 7 and Wave 8 and reported that they do not smoke cigarettes at all nowadays at Wave 8
replace smoking = 1 if heske==3 & heska==2 & wave==8
* Assign the number 0 if the participant reported never having smoked cigarettes and reported that they do not smoke cigarettes at all nowadays at Wave 8
replace smoking = 0 if heske9==1 & heska==2 & wave==8
* Assign the number 1 if the participant reported that they were no longer smoking cigarettes by Wave 8 and reported that they do not smoke cigarettes at all nowadays at Wave 8
replace smoking = 1 if heske9==2 & heska==2 & wave==8

* Assign the number 1 if the participant reported that they stopped smoking between Wave 8 and Wave 9 and reported that they do not smoke cigarettes at all nowadays at Wave 9
replace smoking = 1 if heske==3 & heska==2 & wave==9

* Assign the number 1 if the participant reported that they stopped smoking cigarettes
replace smoking = 1 if heskf==2
* Assign the number 2 if the participant reported smoking cigarettes nowadays
replace smoking = 2 if heskf==1

* Count total number of participants and observations
unique idauniq
* 15,022 individuals, 120,176 observations

* Save dataset with a new name
save datavariables04.dta

* [b45-b48] Cognitive function
* WAVE 2, 3, 4, 5, 6, 7, 8, 9
* [b45] Computed score from date questions (orientation in time)
* Generate a new variable duplicating the cfdatd variable
gen daymonth = cfdatd
```

```
2057  * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2058  replace daymonth = . if daymonth<0
2059  * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2060  replace daymonth = 0 if daymonth==2
2061  * Generate a new variable duplicating the cfday variable
2062  gen day = cfday
2063  * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2064  replace day = . if day<0
2065  * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2066  replace day = 0 if day==2
2067  * Generate a new variable duplicating the cfdaty variable
2068  gen year = cfdaty
2069  * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2070  replace year = . if year<0
2071  * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2072  replace year = 0 if year==2
2073  * Generate a new variable duplicating the cfdatm variable
2074  gen month = cfdatm
2075  * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2076  replace month = . if month<0
2077  * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2078  replace month = 0 if month==2
2079  * Generate a new variable equal to the sum of the four orientation in time items to create a total
      score
2080  gen orientation = daymonth + day + year + month
2081  gen orientation2 = cfdscr if cfdscr>=0
2082  * Generate a new variable and assign the number 0 for participants who scored 0, 1, 2, or 3 points
      on the time orientation test
2083  gen oribi = 0 if inlist(orientation,0,1,2,3)
2084  * Assign the number 1 for participants who answered all questions correctly (i.e., scored 4) on the
      time orientation test
2085  replace oribi = 1 if orientation==4
2086
2087  * WAVE 2, 3, 4, 5, 6, 7, 8, 9
2088  * [b46-b47] Word-list learning (verbal learning and recall)
2089  tab cftest
2090  tab cfwrds
2091  * Generate a new variable duplicating the cflisen variable for participants with a score from 0 to 10
2092  gen learning = cflisen if cflisen>=0
2093  * Generate a new variable duplicating the cflisd variable for participants with a score from 0 to 10
2094  gen recall = cflisd if cflisd>=0
2095
2096  * [b46] Number of words recalled immediately
2097  sum learning
2098  * Assign the number 0 for participants with scores >1 standard deviation below the mean
2099  replace learning = 0 if learning >= 0 & learning < 3.990809
2100  * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2101  replace learning = 1 if learning >= 3.990809 & learning <= 7.640907
2102  * Assign the number 2 for participants with scores >1 standard deviation above the mean
2103  replace learning = 2 if learning > 7.640907 & learning != .
2104
2105  * [b47] Number of words recalled after delay
2106  sum recall
2107  * Assign the number 0 for participants with scores >1 standard deviation below the mean
2108  replace recall = 0 if recall >= 0 & recall < 2.353383
2109  * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2110  replace recall = 1 if recall >= 2.353383 & recall <= 6.666215
2111  * Assign the number 2 for participants with scores >1 standard deviation above the mean
2112  replace recall = 2 if recall > 6.666215 & recall != .
2113
2114  * WAVE 2, 3, 4, 5, 7, 8, 9
2115  * [b48] Number of animals mentioned (verbal fluency)
2116  tab cfani
```

```
* Generate a new variable duplicating the cfani variable for participants with a score of 0 or more
gen fluency = cfani if cfani>=0
sum fluency
* Assign the number 0 for participants with scores >1 standard deviation below the mean
replace fluency = 0 if fluency >= 0 & fluency < 13.60175
* Assign the number 1 for participants with scores ±1 standard deviation around the mean
replace fluency = 1 if fluency >= 13.60175 & fluency <= 27.79781
* Assign the number 2 for participants with scores >1 standard deviation above the mean
replace fluency = 2 if fluency > 27.79781 & fluency != .

* WAVE 2, 4, 6, 8, 9
* [b49] Grip strength
* Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
replace mmgsd1 = . if mmgsd1 < 0
replace mmgsd2 = . if mmgsd2 < 0
replace mmgsd3 = . if mmgsd3 < 0
replace mmgsn1 = . if mmgsn1 < 0
replace mmgsn2 = . if mmgsn2 < 0
replace mmgsn3 = . if mmgsn3 < 0
replace mmgsd1 = . if mmgsd1==99
replace mmgsd2 = . if mmgsd2==99
replace mmgsd3 = . if mmgsd3==99
replace mmgsn1 = . if mmgsn1==99
replace mmgsn2 = . if mmgsn2==99
replace mmgsn3 = . if mmgsn3==99

* Generate a new variable equal to the maximum grip strength across all available measures
egen maxgrip = rowmax(mmgsd1 mmgsd2 mmgsd3 mmgsn1 mmgsn2 mmgsn3)

sum maxgrip
* Assign the number 0 for participants with scores >1 standard deviation below the mean
replace maxgrip = 0 if maxgrip >= 0 & maxgrip < 19.61438
* Assign the number 1 for participants with scores ±1 standard deviation around the mean
replace maxgrip = 1 if maxgrip >= 19.61438 & maxgrip <= 42.36316
* Assign the number 2 for participants with scores >1 standard deviation above the mean
replace maxgrip = 2 if maxgrip > 42.36316 & maxgrip != .

* Save dataset with a new name
save datastop.dta

* [b50-b51] (+ 4 omitted) Biomarkers
* WAVE 2, 4, 6, 8, 9
* [b50] Blood fibrinogen level (g/L)
tab cfib if cfib<0
* Generate a new variable duplicating the cfib variable
gen fibrinogen = cfib
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace fibrinogen = . if fibrinogen < 0
replace fibrinogen = . if fibrinogen > 9000
sum fibrinogen
* Assign the number 1 for participants with a fibrinogen level ≤4 g/L
replace fibrinogen = 1 if fibrinogen <= 4
* Assign the number 0 for participants with a fibrinogen level >4 g/L
replace fibrinogen = 0 if fibrinogen > 4 & fibrinogen != .

* (1 omitted) Blood HDL level (mmol/L)
tab hdl if hdl<0
* Generate a new variable duplicating the hdl variable
gen highdensity = hdl
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace highdensity = . if highdensity < 0
replace highdensity = . if highdensity > 9000
```

```
sum highdensity
* Assign the number 0 for participants with a HDL level <1 mmol/L
replace highdensity = 0 if highdensity <1
* Assign the number 1 for participants with a HDL level ≥1 mmol/L
replace highdensity = 1 if highdensity >= 1 & highdensity != .

* (1 omitted) Blood triglyceride level (mmol/L)
tab trig if trig<0
* Generate a new variable duplicating the trig variable
gen triglyceride = trig
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace triglyceride = . if triglyceride < 0
replace triglyceride = . if triglyceride > 9000
sum triglyceride
* Assign the number 1 for participants with a triglyceride level ≤2 mmol/L
replace triglyceride = 1 if triglyceride <= 2
* Assign the number 0 for participants with a triglyceride level >2 mmol/L
replace triglyceride = 0 if triglyceride > 2 & triglyceride != .

* (1 omitted) Blood LDL level (mmol/L)
tab ldl if ldl<0
* Generate a new variable duplicating the ldl variable
gen lowdensity = ldl
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace lowdensity = . if lowdensity < 0
replace lowdensity = . if lowdensity > 9000
sum lowdensity
* Assign the number 1 for participants with a LDL level ≤4 mmol/L
replace lowdensity = 1 if lowdensity <= 4
* Assign the number 0 for participants with a LDL level >4 mmol/L
replace lowdensity = 0 if lowdensity > 4 & lowdensity != .

* [b51] Blood CRP level (mg/L)
tab hscrp if hscrp<0
* Generate a new variable duplicating the hscrp variable
gen CRP = hscrp
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace CRP = . if CRP < 0
replace CRP = . if CRP > 9000
sum CRP
* Replace variable as missing for CRP values >20 mg/L
replace CRP = . if CRP > 20
* Assign the number 1 for participants with a CRP level ≤3 mg/L
replace CRP = 1 if CRP <= 3
* Assign the number 0 for participants with a CRP level >3 mg/L
replace CRP = 0 if CRP > 3 & CRP != .

* (1 omitted) Blood glycated haemoglobin level (%)
* Wave 2, 4 = %, Wave 6, 8, 9 = mmol/mol
tab hba1c if hba1c<0
* Generate a new variable duplicating the hba1c variable
gen glycated = hba1c
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace glycated = . if glycated < 0
replace glycated = . if glycated > 9000
sum glycated if wave==2
sum glycated if wave==4
sum glycated if wave==6
sum glycated if wave==8
sum glycated if wave==9
* Transform HbA1c levels (mmol/mol) in Wave 6, 8, and 9 into % to match data in Wave 2 and 4
replace glycated = ((glycated/10.929)+2.15) if inlist(wave,6,8,9)
sum glycated if wave==2
```

```
sum glycated if wave==4
sum glycated if wave==6
sum glycated if wave==8
sum glycated if wave==9
* Assign the number 1 for participants with a HbA1c level <6.5 %
replace glycated = 1 if glycated < 6.5
* Assign the number 0 for participants with a HbA1c level ≥6.5 %
replace glycated = 0 if glycated >= 6.5 & glycated != .

* Save dataset with a new name
save alldataefa.dta

********************
***DATA ANALYSIS***
********************

* Keep variables required for analyses
keep idauniq wave revscacta revscactb revscactc revscactd scpt04 scpt05 headldr headlwa headlba
headlea headlbe headlwc headlma headlpr headlsh headlph headlme headlho headlmo hemobwa hemobsi
hemobch hemobcs hemobcl hemobst hemobre hemobpu hemobli hemobpi limiting scorg01 scorg02 scorg03
scorg04 scorg05 scorg06 scorg07 scorg08 balance repcstest gaittest revhehelf psceda pscedb pscedc
pscedd pscede pscedf pscedg pscedh revscfeela revscfeelb revscfeelc oribi learning recall fluency
maxgrip fibrinogen highdensity triglyceride lowdensity CRP glycated
* Rename variables to more convenient forms
rename revscacta scactarev
rename revscactb scactbrev
rename revscactc scactcrev
rename revscactd scactdrev
rename revhehelf hehelfrev
rename revscfeela scfeelarev
rename revscfeelb scfeelbrev
rename revscfeelc scfeelcrev
* Save dataset with a new name
save efanew.dta
* Keep data from Wave 2 only
keep if wave==2
* Drop unnecessary variable
drop wave
* Count total number of participants
unique idauniq
* 15,022 individuals
* Save dataset with a new name
save baseline.dta
* Generate a new variable equal to the sum of variables with available "non-missing" data for each
observation
egen nmcount = rownonmiss(_all), strok
tab nmcount
* Drop observation if >25 % missing values across the 64 variables of interest (idauniq data are
complete for all observations)
drop if nmcount<49
* Drop unnecessary variable
drop nmcount
* Count total number of participants
unique idauniq
* 7,660 individuals
* Generate a new variable equal to the sum of variables with missing data for each observation
egen nmcount = rowmiss(_all)
tab nmcount
* Drop unnecessary variable
drop nmcount
* Set the seed
set seed 1234
* Generate random numbers
```

```
gen random = uniform()
sort random
* Assign 30 % of the total sample to the validation sub-sample
gen byte validation = _n <= 2298
* Save dataset with a new name
save efatotal.dta
* Keep participants assigned to the developmental sub-sample
keep if validation == 0
* Count total number of participants
unique idauniq
* 5,362 individuals
* Save developmental dataset
save developmentalnew.dta
* Use efatotal.dta dataset
use efatotal.dta
* Keep participants assigned to the validation sub-sample
keep if validation == 1
* Count total number of participants
unique idauniq
* 2,298 individuals
* Save validation dataset
save validationnew.dta

* Use developmental dataset
use developmentalnew.dta
* Convert Stata data into a data file and Mplus input file
stata2mplus using developmentalfinal.dta

* Use validation dataset
use validationnew.dta
* Convert Stata data into a data file and Mplus input file
stata2mplus using validationfinal.dta

* Use efanew.dta dataset
use efanew.dta
* Drop variables omitted from final metric
drop pscedb pscedh headlmo scfeelarev scfeelbrev scfeelcrev balance headlme triglyceride lowdensity highdensity glycated scorg06
* Save dataset with a new name
save MLIRT.dta
* Count total number of participants and observations
unique idauniq
* 15,022 individuals, 120,176 observations
* Generate a new variable equal to the sum of variables with available "non-missing" data for each observation
egen nmcount = rownonmiss(_all), strok
tab nmcount
* Drop observation if >50 % missing values across the 51 variables of interest (idauniq and wave data are complete for all observations)
keep if nmcount>=27.5
* Drop unnecessary variable
drop nmcount
* Count total number of participants and observations
unique idauniq
* 14,755 individuals, 66,133 observations
* Generate a new variable equal to the sum of variables with missing data for each observation
egen nmcount = rowmiss(_all)
tab nmcount
* Drop unnecessary variable
drop nmcount
* Save dataset with a new name
save MLIRTtouse.dta
* Export Stata data to .csv file
```

```
export delimited using "", nolabel replace

* Use alldataefa.dta dataset
use alldataefa.dta

* WAVE 2, 3, 4, 5, 6, 8, 9
* Socio-economic covariate - Occupational class
* Excluded Never worked and long-term unemployed
* Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA dataset)
replace nssec8 = . if nssec8<0
replace nssec8 = . if nssec8 == 99
* Generate a new variable
gen mynssec3 = .
* Assign the number 2 if the participant's current or most recent occupation was coded as: Higher managerial, administrative and professional occupations; or Lower managerial, administrative and professional occupations
replace mynssec3 = 2 if inlist(nssec8,1,2)
* Assign the number 1 if the participant's current or most recent occupation was coded as: Intermediate occupation; or Small employers and own account workers
replace mynssec3 = 1 if inlist(nssec8,3,4)
* Assign the number 0 if the participant's current or most recent occupation was coded as: Lower supervisory and technical occupations; or Semi-routine occupations; or Routine occupations
replace mynssec3 = 0 if inlist(nssec8,5,6,7)
* Overwrite dataset, by replacing the previously saved file
save alldataefa.dta, replace

* Keep variables required for analyses
keep idauniq wave indager fqethnr education totwq5_bu_s disex scako smoking activity2 mynssec3 QoL
* Save dataset with a new name
save MLIRTGMM.dta
* One-to-one merge of data in memory with MLIRTdata.dta (exported from RStudio following MLIRT analyses) on participant ID
merge 1:1 idauniq wave using MLIRTdata.dta
* Sort from lowest to highest participant ID
sort idauniq
* Save dataset with a new name
save MLIRTdatafull.dta
tab AHA
* Drop observations with missing AHA scores
drop if AHA==.
* Count total number of participants and observations
unique idauniq
* 14,755 individuals, 66,133 observations
* Save dataset with a new name
save MLIRTfullAHA.dta
sum indager, d
sum indager if wave==2, d
* Keep necessary variables
keep idauniq wave indager fqethnr education totwq5_bu_s disex scako smoking activity2 mynssec3 QoL AHA

* Generate a new variable duplicating the biological sex variable at Wave 2
gen sex = disex if wave==2
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward sex, replace

* Generate a new variable duplicating the education variable at Wave 2
gen qualifications = education if wave==2
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
```

```
2414  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves available for that participant) by participant ID
2415  bysort idauniq: carryforward qualifications, replace
2416
2417  * Generate a new variable duplicating the wealth variable at Wave 2
2418  gen wealth = totwq5_bu_s if wave==2
2419  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2420  tsset idauniq wave
2421  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves available for that participant) by participant ID
2422  bysort idauniq: carryforward wealth, replace
2423
2424  * Generate a new variable duplicating the age variable at Wave 2
2425  gen age = indager if wave==2
2426  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2427  tsset idauniq wave
2428  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves available for that participant) by participant ID
2429  bysort idauniq: carryforward age, replace
2430
2431  * Generate a new variable duplicating the ethnicity variable at Wave 2
2432  gen ethnicity = fqethnr if wave==2
2433  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2434  tsset idauniq wave
2435  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves available for that participant) by participant ID
2436  bysort idauniq: carryforward ethnicity, replace
2437
2438  * Generate a new variable duplicating the occupational class variable at Wave 2
2439  gen nssec3 = mynssec3 if wave==2
2440  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2441  tsset idauniq wave
2442  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 2 to the
      follow-up waves available for that participant) by participant ID
2443  bysort idauniq: carryforward nssec3, replace
2444
2445  * Save dataset with a new name
2446  save GMMcovdataSES0317.dta
2447  * One-to-one merge of data in memory with data.dta on participant ID and wave
2448  merge 1:1 idauniq wave using data.dta, generate (merge_finstat)
2449  * Keep if matched
2450  keep if merge_finstat==3
2451
2452  tab sex if finstatw3=="C3CM"
2453  tab disex if finstatw3=="C3CM"
2454  * Generate a new variable duplicating the sex variable for the Wave 3 refreshment sample
2455  gen sex3 = disex if finstatw3=="C3CM"
2456  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2457  tsset idauniq wave
2458  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the
      follow-up waves available for that participant) by participant ID
2459  bysort idauniq: carryforward sex3, replace
2460
2461  tab sex if finstatw4=="C4CM"
2462  tab disex if finstatw4=="C4CM"
2463  * Generate a new variable duplicating the sex variable for the Wave 4 refreshment sample
2464  gen sex4 = disex if finstatw4=="C4CM"
2465  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2466  tsset idauniq wave
2467  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the
      follow-up waves available for that participant) by participant ID
2468  bysort idauniq: carryforward sex4, replace
2469
```

```
2470  tab sex if finstatw6==25
2471  tab disex if finstatw6==25
2472  * Generate a new variable duplicating the sex variable for the Wave 6 refreshment sample
2473  gen sex6 = disex if finstatw6==25
2474  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2475  tsset idauniq wave
2476  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the
      follow-up waves available for that participant) by participant ID
2477  bysort idauniq: carryforward sex6, replace
2478
2479  tab sex if finstatw7==33
2480  tab disex if finstatw7==33
2481  * Generate a new variable duplicating the sex variable for the Wave 7 refreshment sample
2482  gen sex7 = disex if finstatw7==33
2483  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2484  tsset idauniq wave
2485  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the
      follow-up waves available for that participant) by participant ID
2486  bysort idauniq: carryforward sex7, replace
2487
2488  tab sex if finstatw9==48
2489  tab indsex if finstatw9==48
2490  * Assign the number 0 if the participant is male at Wave 9
2491  gen indsex9 = 0 if indsex==1 & wave==9
2492  * Assign the number 1 if the participant is female at Wave 9
2493  replace indsex9 = 1 if indsex==2 & wave==9
2494  * Generate a new variable duplicating the sex variable for the Wave 9 refreshment sample
2495  gen sex9 = indsex9 if finstatw9==48
2496
2497  * Generate a new variable duplicating the time-constant sex variable (assessed at Wave 2)
2498  gen sexcons = sex
2499  * Replace the sex variable with values from the first wave of data collection for the refreshment
      samples
2500  replace sexcons = sex3 if sexcons==. & sex3!=.
2501  replace sexcons = sex4 if sexcons==. & sex4!=.
2502  replace sexcons = sex6 if sexcons==. & sex6!=.
2503  replace sexcons = sex7 if sexcons==. & sex7!=.
2504  replace sexcons = sex9 if sexcons==. & sex9!=.
2505
2506  tab ethnicity if finstatw3=="C3CM"
2507  tab fqethnr if finstatw3=="C3CM"
2508  * Generate a new variable duplicating the ethnicity variable for the Wave 3 refreshment sample
2509  gen eth3 = fqethnr if finstatw3=="C3CM"
2510  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2511  tsset idauniq wave
2512  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the
      follow-up waves available for that participant) by participant ID
2513  bysort idauniq: carryforward eth3, replace
2514
2515  tab ethnicity if finstatw4=="C4CM"
2516  tab fqethnr if finstatw4=="C4CM"
2517  * Generate a new variable duplicating the ethnicity variable for the Wave 4 refreshment sample
2518  gen eth4 = fqethnr if finstatw4=="C4CM"
2519  * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2520  tsset idauniq wave
2521  * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the
      follow-up waves available for that participant) by participant ID
2522  bysort idauniq: carryforward eth4, replace
2523
2524  tab ethnicity if finstatw6==25
2525  tab fqethnr if finstatw6==25
2526  * Generate a new variable duplicating the ethnicity variable for the Wave 6 refreshment sample
2527  gen eth6 = fqethnr if finstatw6==25
```

```
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward eth6, replace

tab ethnicity if finstatw7==33
tab fqethnr if finstatw7==33
* Generate a new variable duplicating the ethnicity variable for the Wave 7 refreshment sample
gen eth7 = fqethnr if finstatw7==33
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward eth7, replace

tab ethnicity if finstatw9==48
tab fqethnr if finstatw9==48
* Generate a new variable duplicating the ethnicity variable for the Wave 9 refreshment sample
gen eth9 = fqethnr if finstatw9==48

* Generate a new variable duplicating the time-constant ethnicity variable (assessed at Wave 2)
gen ethcons = ethnicity
* Replace the ethnicity variable with values from the first wave of data collection for the refreshment samples
replace ethcons = eth3 if ethcons==. & eth3!=.
replace ethcons = eth4 if ethcons==. & eth4!=.
replace ethcons = eth6 if ethcons==. & eth6!=.
replace ethcons = eth7 if ethcons==. & eth7!=.
replace ethcons = eth9 if ethcons==. & eth9!=.

tab qualifications if finstatw3=="C3CM"
tab education if finstatw3=="C3CM"
* Generate a new variable duplicating the education variable for the Wave 3 refreshment sample
gen edu3 = education if finstatw3=="C3CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward edu3, replace

tab qualifications if finstatw4=="C4CM"
tab education if finstatw4=="C4CM"
* Generate a new variable duplicating the education variable for the Wave 4 refreshment sample
gen edu4 = education if finstatw4=="C4CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward edu4, replace

tab qualifications if finstatw6==25
tab education if finstatw6==25
* Generate a new variable duplicating the education variable for the Wave 6 refreshment sample
gen edu6 = education if finstatw6==25
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward edu6, replace

tab qualifications if finstatw7==33
tab education if finstatw7==33
```

```
* Generate a new variable duplicating the education variable for the Wave 7 refreshment sample
gen edu7 = education if finstatw7==33
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward edu7, replace

tab qualifications if finstatw9==48
tab education if finstatw9==48
* Generate a new variable duplicating the education variable for the Wave 9 refreshment sample
gen edu9 = education if finstatw9==48

* Generate a new variable duplicating the time-constant education variable (assessed at Wave 2)
gen educons = qualifications
* Replace the education variable with values from the first wave of data collection for the refreshment samples
replace educons = edu3 if educons==. & edu3!=.
replace educons = edu4 if educons==. & edu4!=.
replace educons = edu6 if educons==. & edu6!=.
replace educons = edu7 if educons==. & edu7!=.
replace educons = edu9 if educons==. & edu9!=.

tab nssec3 if finstatw3=="C3CM"
tab mynssec3 if finstatw3=="C3CM"
* Generate a new variable duplicating the occupational class variable for the Wave 3 refreshment sample
gen occ3 = mynssec3 if finstatw3=="C3CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward occ3, replace

tab nssec3 if finstatw4=="C4CM"
tab mynssec3 if finstatw4=="C4CM"
* Generate a new variable duplicating the occupational class variable for the Wave 4 refreshment sample
gen occ4 = mynssec3 if finstatw4=="C4CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward occ4, replace

tab nssec3 if finstatw6==25
tab mynssec3 if finstatw6==25
* Generate a new variable duplicating the occupational class variable for the Wave 6 refreshment sample
gen occ6 = mynssec3 if finstatw6==25
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward occ6, replace
* Save dataset with a new name
save intermediate.dta
clear

* Recode the long form occupational class variable at Wave 7 into the three-class classification
use idauniq NSSEC using wave_7_elsa_data.dta
* Generate a new variable called wave and assign the number 7 to each observation (to designate Wave 7)
```

```
gen wave = 7
* Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
replace NSSEC = . if NSSEC==-1
* Replace variable with the number 0 if the participant's current or most recent occupation was coded as: Lower supervisory and technical occupations; or Semi-routine occupations; or Routine occupations
replace NSSEC = 0 if inlist(NSSEC,10,11.1,11.2,12.1,12.2,12.3,12.4,12.5,12.6,12.7,13.1,13.2,13.3,13.4,13.5)
* Replace variable with the number 1 if the participant's current or most recent occupation was coded as: Intermediate occupation; or Small employers and own account workers
replace NSSEC = 1 if inlist(NSSEC,9.1,9.2,8.1,8.2,7.1,7.2,7.3,7.4)
* Replace variable with the number 2 if the participant's current or most recent occupation was coded as: Higher managerial, administrative and professional occupations; or Lower managerial, administrative and professional occupations
replace NSSEC = 2 if inlist(NSSEC,3.1,3.2,3.3,3.4,4.1,4.2,4.3,4.4,6,5)
* Save dataset with a new name
save wave7nssec.dta

* Use dataset for creating the time-constant variables
use intermediate.dta
* Drop the long form occupational class variable
drop NSSEC
* One-to-one merge of data in memory with wave7nssec.dta on participant ID and wave
merge 1:1 idauniq wave using wave7nssec.dta, generate (merge_nssec)
* Restore sample size
keep if AHA!=.

tab nssec3 if finstatw7==33
tab NSSEC if finstatw7==33 & wave==7
* Generate a new variable duplicating the occupational class variable for the Wave 7 refreshment sample
gen occ7 = NSSEC if finstatw7==33
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward occ7, replace

tab nssec3 if finstatw9==48
tab mynssec3 if finstatw9==48
* Generate a new variable duplicating the occupational class variable for the Wave 9 refreshment sample
gen occ9 = mynssec3 if finstatw9==48

* Generate a new variable duplicating the time-constant occupational class variable (assessed at Wave 2)
gen occcons = nssec3
* Replace the occupational class variable with values from the first wave of data collection for the refreshment samples
replace occcons = occ3 if occcons==. & occ3!=.
replace occcons = occ4 if occcons==. & occ4!=.
replace occcons = occ6 if occcons==. & occ6!=.
replace occcons = occ7 if occcons==. & occ7!=.
replace occcons = occ9 if occcons==. & occ9!=.

tab wealth if finstatw3=="C3CM"
tab totwq5_bu_s if finstatw3=="C3CM"
* Generate a new variable duplicating the wealth variable for the Wave 3 refreshment sample
gen wealth3 = totwq5_bu_s if finstatw3=="C3CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the follow-up waves available for that participant) by participant ID
```

```
bysort idauniq: carryforward wealth3, replace

tab wealth if finstatw4=="C4CM"
tab totwq5_bu_s if finstatw4=="C4CM"
* Generate a new variable duplicating the wealth variable for the Wave 4 refreshment sample
gen wealth4 = totwq5_bu_s if finstatw4=="C4CM"
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward wealth4, replace

tab wealth if finstatw6==25
tab totwq5_bu_s if finstatw6==25
* Generate a new variable duplicating the wealth variable for the Wave 6 refreshment sample
gen wealth6 = totwq5_bu_s if finstatw6==25
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward wealth6, replace

tab wealth if finstatw7==33
tab totwq5_bu_s if finstatw7==33
* Generate a new variable duplicating the wealth variable for the Wave 7 refreshment sample
gen wealth7 = totwq5_bu_s if finstatw7==33
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward wealth7, replace

tab wealth if finstatw9==48
tab totwq5_bu_s if finstatw9==48
* Generate a new variable duplicating the wealth variable for the Wave 9 refreshment sample
gen wealth9 = totwq5_bu_s if finstatw9==48

* Generate a new variable duplicating the time-constant wealth variable (assessed at Wave 2)
gen wealthcons = wealth
* Replace the wealth variable with values from the first wave of data collection for the refreshment samples
replace wealthcons = wealth3 if wealthcons==. & wealth3!=.
replace wealthcons = wealth4 if wealthcons==. & wealth4!=.
replace wealthcons = wealth6 if wealthcons==. & wealth6!=.
replace wealthcons = wealth7 if wealthcons==. & wealth7!=.
replace wealthcons = wealth9 if wealthcons==. & wealth9!=.

tab age if finstatw3=="C3CM"
tab indager if finstatw3=="C3CM"
* Generate a new variable duplicating the age variable for the Wave 3 refreshment sample
gen agew3 = indager if finstatw3=="C3CM" & wave==3

tab age if finstatw4=="C4CM"
tab indager if finstatw4=="C4CM"
* Generate a new variable duplicating the age variable for the Wave 4 refreshment sample
gen agew4 = indager if finstatw4=="C4CM" & wave==4

tab age if finstatw6==25
tab indager if finstatw6==25
* Generate a new variable duplicating the age variable for the Wave 6 refreshment sample
gen agew6 = indager if finstatw6==25 & wave==6

tab age if finstatw7==33
```

```
tab indager if finstatw7==33
* Generate a new variable duplicating the age variable for the Wave 7 refreshment sample
gen agew7 = indager if finstatw7==33 & wave==7

tab age if finstatw9==48
tab indager if finstatw9==48
* Generate a new variable duplicating the age variable for the Wave 9 refreshment sample
gen agew9 = indager if finstatw9==48 & wave==9

* Generate a new variable duplicating the age variable for the Wave 3 refreshment sample
gen agew3cons = agew3 if wave==3
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 3 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward agew3cons, replace

* Generate a new variable duplicating the age variable for the Wave 4 refreshment sample
gen agew4cons = agew4 if wave==4
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 4 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward agew4cons, replace

* Generate a new variable duplicating the age variable for the Wave 6 refreshment sample
gen agew6cons = agew6 if wave==6
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward agew6cons, replace

* Generate a new variable duplicating the age variable for the Wave 7 refreshment sample
gen agew7cons = agew7 if wave==7
* Declare a panel dataset with participant ID "idauniq" and time variable "wave"
tsset idauniq wave
* Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the follow-up waves available for that participant) by participant ID
bysort idauniq: carryforward agew7cons, replace

* Generate a new variable duplicating the time-constant age variable (assessed at Wave 2)
gen agewcons = age
* Replace the age variable with values from the first wave of data collection for the refreshment samples
replace agewcons = agew3cons if agewcons==. & agew3cons!=.
replace agewcons = agew4cons if agewcons==. & agew4cons!=.
replace agewcons = agew6cons if agewcons==. & agew6cons!=.
replace agewcons = agew7cons if agewcons==. & agew7cons!=.
replace agewcons = agew9 if agewcons==. & agew9!=.

* Save dataset with a new name
save timeconstant.dta

* Keep variables required for analyses
keep idauniq wave sexcons educons wealthcons agewcons ethcons occcons QoL AHA
* Reshape data into wide format for observations identified by participant ID and add wave as an identifying time period
reshape wide QoL AHA, j(wave) i(idauniq)
* Save dataset with a new name
save GMMcovdatawideSES0323.dta
sum QoL9, d
* Generate a new variable and assign the number 0 if a participant's quality-of-life score is below
```

```
the sample median
gen QoL9binary = 0 if QoL9<43
* Assign the number 1 if a participant's quality-of-life score is above or equal to the sample median
replace QoL9binary = 1 if QoL9 >=43 & QoL9!=.
* Save dataset with a new name
save GMMcovbinarySES0323.dta

* Dummy variables for conditional GMM
* Education
* Medium education (i.e., school qualifications) (coded as 1) versus low (i.e., no formal qualifications) or high (i.e., higher education) education (coded as 0)
gen mediumed = 0 if inlist(educons,0,2)
replace mediumed = 1 if educons == 1
* High education (coded as 1) versus low or medium education (coded as 0)
gen highed = 0 if inlist(educons,0,1)
replace highed = 1 if educons == 2
* Occupational class
* Intermediate occupations (coded as 1) versus lower or higher occupations (coded as 0)
gen mediumocc = 0 if inlist(occcons,0,2)
replace mediumocc = 1 if occcons == 1
* Higher occupations (coded as 1) versus lower or intermediate occupations (coded as 0)
gen highocc = 0 if inlist(occcons,0,1)
replace highocc = 1 if occcons == 2
* Wealth
* 2nd quintile (coded as 1) versus 1st, 3rd, 4th, or 5th quintile (coded as 0)
gen quint2 = 0 if inlist(wealthcons,1,3,4,5)
replace quint2 = 1 if wealthcons == 2
* 3rd quintile (coded as 1) versus 1st, 2nd, 4th, or 5th quintile (coded as 0)
gen quint3 = 0 if inlist(wealthcons,1,2,4,5)
replace quint3 = 1 if wealthcons == 3
* 4th quintile (coded as 1) versus 1st, 2nd, 3rd, or 5th quintile (coded as 0)
gen quint4 = 0 if inlist(wealthcons,1,2,3,5)
replace quint4 = 1 if wealthcons == 4
* 5th quintile (coded as 1) versus 1st, 2nd, 3rd, or 4th quintile (coded as 0)
gen quint5 = 0 if inlist(wealthcons,1,2,3,4)
replace quint5 = 1 if wealthcons == 5
* Save dataset with a new name
save GMMcovbinarySESdu0323.dta
* Convert Stata data into a data file and Mplus input file
stata2mplus using GMMcovbinarySESdu0323.dta
clear

* Import posterior probabilities of class membership and most likely class membership following step 1 of the three-step manual GMM procedure
import excel "", sheet("") firstrow
* Convert Stata data into a data file and Mplus input file
stata2mplus using step2c3manual0323
* Save dataset with a new name
save step2c3manual0323.dta

* Summarise the AHA scores at each wave
sum AHA2, d
sum AHA3, d
sum AHA4, d
sum AHA5, d
sum AHA6, d
sum AHA7, d
sum AHA8, d
sum AHA9, d

* Summarise the posterior probabilities for each latent class
sum c1 if n==1
sum c2 if n==2
```

```
sum c3 if n==3
clear

* Use timeconstant.dta dataset
use timeconstant.dta
* Count total number of participants and observations
unique idauniq
* 14,755 individuals, 66,133 observations
* Drop unnecessary variables
drop obsnr obscount
* Generate a variable that assigns the observation number (i.e., 1 for first data collection timepoint, 2 for second data collection timepoint etc.) to each row by participant ID
bysort idauniq (wave): gen obsnr = _n
* Generate a variable that assigns the number of total observations to each row of data for a given participant
bysort idauniq: gen obscount = _N
tab wave
* Summarise number of observations per participant
tab obscount if obsnr==1
sum obscount if obsnr==1
* Keep variables required for Receiver Operating Characteristic (ROC) analyses
keep idauniq wave sexcons QoL AHA
* Save dataset with a new name
save ROCdata0323.dta
sum QoL, d
* Generate a new variable and assign the number 0 if a participant's quality-of-life score is below the sample median
gen QoLbinary = 0 if QoL < 43
* Assign the number 1 if a participant's quality-of-life score is above or equal to the sample median
replace QoLbinary = 1 if QoL >= 43 & QoL !=.
* Overwrite dataset, by replacing the previously saved file
save ROCdata0323.dta, replace
* Drop unnecessary variable
drop QoL
* Reshape data into wide format for observations identified by participant ID and add wave as an identifying time period
reshape wide AHA QoLbinary, j(wave) i (idauniq)
* Save dataset with a new name
save ROC92wide0323.dta

* ROC analyses, clustering at the participant level and adjusting the control distribution for biological sex
* Quality-of-life at Wave 3
rocreg QoLbinary3 AHA2, probit ml ctrlcov(sexcons) ctrlmodel(linear) cluster(idauniq)
* Quality-of-life at Wave 6
rocreg QoLbinary6 AHA2, probit ml ctrlcov(sexcons) ctrlmodel(linear) cluster(idauniq)
* Quality-of-life at Wave 9
rocreg QoLbinary9 AHA2, probit ml ctrlcov(sexcons) ctrlmodel(linear) cluster(idauniq)
clear

* Use timeconstant.dta dataset
use timeconstant.dta
* Two-level mixed-effects linear regression of AHA scores on lifestyle behaviours (adjusted for covariates), with random intercepts by participant ID
mixed AHA indager i.sexcons i.ethcons i.scako i.smoking i.activity2 || idauniq:
* Store estimates for later use
estimates store randint
* Fit indices
estimates stats
* Two-level mixed-effects linear regression of AHA scores on lifestyle behaviours (adjusted for covariates), with random intercepts by participant ID and a random slope according to wave
mixed AHA indager i.sexcons i.ethcons i.scako i.smoking i.activity2 || idauniq: wave
* Store estimates for later use
```

```
estimates store randslope
* Fit indices
estimates stats
* Likelihood ratio test
lrtest randslope randint
clear

* Use GMMcovbinarySES0323.dta dataset
use GMMcovbinarySES0323.dta
* Drop observations with missing data on socio-economic or demographic covariates
drop if sexcons==. | educons==. | wealthcons==. | agewcons==. | ethcons==. | occcons ==.
* Count total number of participants and observations
unique idauniq
* 11,566 individuals
* Keep necessary variables
keep idauniq sexcons educons wealthcons agewcons ethcons occcons
* Save dataset with a new name
save descbyclassGMMcovdataSESwide0325.dta
clear

* Import posterior probabilities of class membership and most likely class membership following step 3 of the three-step manual GMM procedure
import excel "", sheet("") firstrow
* Save dataset with a new name
save manualstep3c3.dta
clear

* Use descbyclassGMMcovdataSESwide0325.dta dataset
use descbyclassGMMcovdataSESwide0325.dta
* One-to-one merge of data in memory with manualstep3c3.dta on participant ID
merge 1:1 idauniq using manualstep3c3.dta, generate (merge_posterior)
* Overwrite dataset, by replacing the previously saved file
save descbyclassGMMcovdataSESwide0325.dta, replace

* Descriptive statistics stratified by class membership
sum agewcons if n==1
sum agewcons if n==2
sum agewcons if n==3

tab sexcons if n==1
tab sexcons if n==2
tab sexcons if n==3

tab ethcons if n==1
tab ethcons if n==2
tab ethcons if n==3

tab educons if n==1
tab educons if n==2
tab educons if n==3

tab occcons if n==1
tab occcons if n==2
tab occcons if n==3

tab wealthcons if n==1
tab wealthcons if n==2
tab wealthcons if n==3
```