

```

1 *****
2 ***SYNTAX FOR "Socio-economic inequalities 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 skako
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
82  2)
83  gen wave = 2
84  * Save Wave 2 core dataset
85  save wave2.dta
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

```

```

94  rename sampsta samptyp3
95  rename w3edqual edqual
96  rename finstat finstatw3
97  * Generate a new variable called wave and assign the number 3 to each observation (to designate Wave
    3)
98  gen wave = 3
99  * Save Wave 3 core dataset
100 save wave3.dta
101
102 * Variables Wave 4
103 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
104 * Describe dataset
105 describe
106 * Sort from lowest to highest participant ID
107 sort idauniq
108 * Rename variables to ensure consistency across waves
109 rename w4nssec8 nssec8
110 rename headlte headlph
111 rename scpt04 scpt05
112 rename scpt03 scpt04
113 rename w4edqual edqual
114 rename samptyp samptyp4
115 rename finstat4 finstatw4
116 * Generate a new variable called wave and assign the number 4 to each observation (to designate Wave
    4)
117 gen wave = 4
118 * Save Wave 4 core dataset
119 save wave4.dta
120
121 * Variables Wave 5
122 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
123 * Describe dataset
124 describe
125 * Sort from lowest to highest participant ID
126 sort idauniq
127 * Rename variables to ensure consistency across waves
128 rename w5nssec8 nssec8
129 rename headlte headlph
130 rename scpt04 scpt05
131 rename scpt03 scpt04
132 rename w5edqual edqual
133 rename samptyp samptyp5
134 * Generate a new variable called wave and assign the number 5 to each observation (to designate Wave
    5)
135 gen wave = 5

```

```

136 * Save Wave 5 core dataset
137 save wave5.dta
138
139 * Variables Wave 6
140 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
141 * Describe dataset
142 describe
143 * Sort from lowest to highest participant ID
144 sort idauniq
145 * Rename variables to ensure consistency across waves
146 rename MmSchs mmschs
147 rename MmWill mmwill
148 rename MmSaf mmsaf
149 rename MmAvsp mmavsp
150 rename MmWala mmwala
151 rename MmAid mmaid
152 rename CfDatD cfdatd
153 rename CfDatM cfdatm
154 rename CfDatY cfdaty
155 rename CfDay cfday
156 rename CfTest cftest
157 rename CfWrds cfwrds
158 rename w6nssec8 nssec8
159 rename Heill heill
160 rename Helim helim
161 rename MmAlone mmalone
162 rename MmHSS mmhss
163 rename Hehelf hehelf
164 rename MmTrya mmtrya
165 rename MMWlKA mmwlka
166 rename MmTryb mmtryb
167 rename MMWlKB mmwlkb
168 rename scptr3 scpt04
169 rename scptr4 scpt05
170 rename DhSex dhsex
171 rename DiSex disex
172 rename HeSmk hesmk
173 rename HESka heska
174 rename HeSkb heskb
175 rename HeSkc heskc
176 rename HeSkd heskd
177 rename HeSke heske
178 rename HeSkf heskf
179 rename HeActa heacta
180 rename HeActb heactb
181 rename HeActc heactc
182 rename Fqethnr fqethnr
183 rename samptyp samtyp6
184 rename PScedA psceda
185 rename PScedB pscedb
186 rename PScedC pscedc
187 rename PScedD pscedd
188 rename PScedE pscede
189 rename PScedF pscedf
190 rename PScedG pscedg

```

```

191  rename PScedH pscedh
192  rename CfDScr cfdscr
193  rename CfLisEn cflisen
194  rename CfLisD cflisd
195  * Generate a new variable called wave and assign the number 6 to each observation (to designate Wave
    6)
196  gen wave = 6
197  * Save Wave 6 core dataset
198  save wave6.dta
199
200  * Variables Wave 7
201  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
202  * Describe dataset
203  describe
204  * Sort from lowest to highest participant ID
205  sort idauniq
206  * Rename variables to ensure consistency across waves
207  rename MmSchs mmschs
208  rename MmWill mmwill
209  rename MmSaf mmsaf
210  rename MmAvsp mmavsp
211  rename MmWala mmwala
212  rename MmAid mmaid
213  rename CfDatD cfdatd
214  rename CfDatM cfdatm
215  rename CfDatY cfdaty
216  rename CfDay cfday
217  rename CfTest cftest
218  rename CfWrds cfwrds
219  rename Heill heill
220  rename Helim helim
221  rename MmAlone mmalone
222  rename MmHSS mmhss
223  rename Hehelf hehelf
224  rename MmTrya mmtrya
225  rename MMWlKA mmwlka
226  rename MmTryb mmtryb
227  rename MMWlKB mmwlkb
228  rename scptr3 scpt04
229  rename scptr4 scpt05
230  rename DhSex dhsex
231  rename DiSex disex
232  rename HeSmk hesmk
233  rename HESka heska
234  rename HeSkb heskb
235  rename HeSkc heskc
236  rename HeSkd heskd
237  rename HeSke heske
238  rename HeSkf heskf
239  rename HeActa heacta
240  rename HeActb heactb
241  rename HeActc heactc
242  rename Fqethnr fqethnr
243  rename samptyp samptyp7
244  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 hesk 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 hesk 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

```

```

290 describe
291 * Sort from lowest to highest participant ID
292 sort idauniq
293 * Rename variables to ensure consistency across waves
294 rename w9nssec8 nssec8
295 rename scorgpo scorg01
296 rename scorgnw scorg02
297 rename scorgrl scorg03
298 rename scorgch scorg04
299 rename scorged scorg05
300 rename scorgsc scorg06
301 rename scorgsp scorg07
302 rename scorg95 scorg08
303 rename scptruk scpt04
304 rename scptrab scpt05
305 rename scalcm scako
306 rename fqethnmr fqethnr
307 rename samptyp samptyp9
308 rename finstat finstatw9
309 * Generate a new variable called wave and assign the number 9 to each observation (to designate Wave
310 9)
311 gen wave = 9
312 * Save Wave 9 core dataset
313 save wave9.dta
314
315 * Variables Wave 2 Financial Derived
316 use idauniq totwq5_bu_s using wave_2_financial_derived_variables.dta
317 * Describe dataset
318 describe
319 * Sort from lowest to highest participant ID
320 sort idauniq
321 * Save Wave 2 financial derived dataset
322 save wave2financial.dta
323
324 * Variables Wave 2 Nurse
325 use idauniq mms sre mmssti mmstre mmstti mmftre2 mmrrre mmftti mmrrfti mmsna mmstna mmftna mmcrre
326 mmcrna mmrrna cfib hdl trig ldl hscrp hba1c mmgsd1 mmgsn1 mmgsd2 mmgsn2 mmgsd3 mmgsn3 mmgswil mmgsdom
327 mmgssta mmgstp mmgsres mmbcsc mmsssc mmstsc mmftsc mmcrav mmcrsc mmrrsc mmrrtti mmrroc using
328 wave_2_nurse_data_v2.dta
329 * Describe dataset
330 describe
331 * Sort from lowest to highest participant ID
332 sort idauniq
333 * Save Wave 2 nurse dataset
334 save wave2nurse.dta
335
336 * Variables Wave 2 Derived
337 use idauniq w2edqual using wave_2_derived_variables.dta
338 * Describe dataset
339 describe
340 * Sort from lowest to highest participant ID
341 sort idauniq
342 * Rename variables to shorter or more convenient forms
343 rename w2edqual edqual
344 * Save Wave 2 derived dataset
345 save wave2derived.dta
346
347 * Variables Wave 3 Financial Derived
348 use idauniq totwq5_bu_s using wave_3_financial_derived_variables.dta
349 * Describe dataset
350 describe
351 * Sort from lowest to highest participant ID
352 sort idauniq

```







```

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

```

```

468 describe
469 * Sort from lowest to highest participant ID
470 sort idauniq
471 * Save Wave 9 financial derived dataset
472 save wave9financial.dta
473
474 * Wave 9 Derived variables
475 use idauniq edqual using wave_9_ifs_derived_variables.dta
476 * Describe dataset
477 describe
478 * Sort from lowest to highest participant ID
479 sort idauniq
480 * Save Wave 9 derived dataset
481 save wave9derived.dta
482
483 * Wave 2 complete data
484 * Merge core, financial, nurse, and derived datasets for Wave 2 using the participant ID
485 * Use Wave 2 core dataset
486 use wave2.dta
487 * One-to-one merge of data in memory with wave2financial.dta on participant ID
488 merge 1:1 idauniq using wave2financial.dta, generate (merge_financial2)
489 * Overwrite Wave 2 dataset, by replacing the previously saved file
490 save wave2.dta, replace
491 * Use the newly saved file for Wave 2
492 use wave2.dta
493 * One-to-one merge of data in memory with wave2nurse.dta on participant ID
494 merge 1:1 idauniq using wave2nurse.dta, generate (merge_nurse2)
495 * Overwrite Wave 2 dataset, by replacing the previously saved file
496 save wave2.dta, replace
497 * Use the newly saved file for Wave 2
498 use wave2.dta
499 * One-to-one merge of data in memory with wave2derived.dta on participant ID
500 merge 1:1 idauniq using wave2derived.dta, generate (merge_derived2)
501 * Sort from lowest to highest participant ID
502 sort idauniq
503 * Overwrite Wave 2 dataset, by replacing the previously saved file
504 save wave2.dta, replace
505
506 * Wave 3 complete data
507 * Merge core and financial datasets for Wave 3 using the participant ID
508 * Use Wave 3 core dataset
509 use wave3.dta
510 * One-to-one merge of data in memory with wave3financial.dta on participant ID
511 merge 1:1 idauniq using wave3financial.dta, generate (merge_financial3)
512 * Sort from lowest to highest participant ID
513 sort idauniq
514 * Overwrite Wave 3 dataset, by replacing the previously saved file
515 save wave3.dta, replace
516
517 * Wave 4 complete data
518 * Merge core, financial, and nurse datasets for Wave 4 using the participant ID
519 * Use Wave 4 core dataset
520 use wave4.dta
521 * One-to-one merge of data in memory with wave4financial.dta on participant ID
522 merge 1:1 idauniq using wave4financial.dta, generate (merge_financial4)
523 * Overwrite Wave 4 dataset, by replacing the previously saved file
524 save wave4.dta, replace
525 * Use the newly saved file for Wave 4
526 use wave4.dta
527 * One-to-one merge of data in memory with wave4nurse.dta on participant ID
528 merge 1:1 idauniq using wave4nurse.dta, generate (merge_nurse4)
529 * Sort from lowest to highest participant ID
530 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

```

```

594 use wave8.dta
595 * One-to-one merge of data in memory with wave8nurse.dta on participant ID
596 merge 1:1 idauniq using wave8nurse.dta, generate (merge_nurse8)
597 * Sort from lowest to highest participant ID
598 sort idauniq
599 * Overwrite Wave 8 dataset, by replacing the previously saved file
600 save wave8.dta, replace
601 * Use the newly saved file for Wave 8
602 use wave8.dta
603 * One-to-one merge of data in memory with wave8derived.dta on participant ID
604 merge 1:1 idauniq using wave8derived.dta, generate (merge_derived8)
605 * Sort from lowest to highest participant ID
606 sort idauniq
607 * Overwrite Wave 8 dataset, by replacing the previously saved file
608 save wave8.dta, replace
609
610 * Wave 9 complete data
611 * Merge core, financial, nurse, and derived datasets for Wave 9 using the participant ID
612 * Use Wave 9 core dataset
613 use wave9.dta
614 * One-to-one merge of data in memory with wave9financial.dta on participant ID
615 merge 1:1 idauniq using wave9financial.dta, generate (merge_financial9)
616 * Overwrite Wave 9 dataset, by replacing the previously saved file
617 save wave9.dta, replace
618 * Use the newly saved file for Wave 9
619 use wave9.dta
620 * One-to-one merge of data in memory with wave9derived.dta on participant ID
621 merge 1:1 idauniq using wave9derived.dta, generate (merge_derived9)
622 * Sort from lowest to highest participant ID
623 sort idauniq
624 * Overwrite Wave 9 dataset, by replacing the previously saved file
625 save wave9.dta, replace
626 * Use the newly saved file for Wave 9
627 use wave9.dta
628 * One-to-one merge of data in memory with wave9nurse.dta on participant ID
629 merge 1:1 idauniq using wave9nurse.dta, generate (merge_nurse9)
630 * Sort from lowest to highest participant ID
631 sort idauniq
632 * Overwrite Wave 9 dataset, by replacing the previously saved file
633 save wave9.dta, replace
634
635 * Append Wave 3 dataset to Wave 2 dataset
636 use wave2.dta
637 append using wave3.dta
638 * Sort by participant ID and wave (lowest to highest)
639 sort idauniq wave
640 * Append Wave 4 dataset
641 append using wave4.dta
642 * Sort by participant ID and wave (lowest to highest)
643 sort idauniq wave
644 * Append Wave 5 dataset
645 append using wave5.dta
646 * Sort by participant ID and wave (lowest to highest)
647 sort idauniq wave
648 * Append Wave 6 dataset
649 append using wave6.dta
650 * Sort by participant ID and wave (lowest to highest)
651 sort idauniq wave
652 * Append Wave 7 dataset
653 append using wave7.dta
654 * Sort by participant ID and wave (lowest to highest)
655 sort idauniq wave
656 * Append Wave 8 dataset

```

```

657 append using wave8.dta
658 * Sort by participant ID and wave (lowest to highest)
659 sort idauniq wave
660 * Append Wave 9 dataset
661 append using wave9.dta
662 * Sort by participant ID and wave (lowest to highest)
663 sort idauniq wave
664 * Assign a number in ascending order to each row of observations
665 gen ascnr = _n
666
667 * Organising dataset
668 * 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
669 bysort idauniq (wave): gen obsnr = _n
670 * Generate a variable that assigns the number of total observations to each row of data for a given
participant
671 bysort idauniq: gen obscount = _N
672 * 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
673 tabulate obscount if obsnr==1
674 * Generate a variable that assigns the number 1 to the row representing participants' first
observation
675 bysort idauniq (wave): gen first = 1 if _n==1
676 * Generate a variable that assigns the number 1 to the row representing participants' last observation
677 bysort idauniq (wave): gen last = 1 if _n==_N
678 * Generate a variable that assigns the number 1 to the row representing participants' first
observation if this corresponds to Wave 2 (baseline)
679 bysort idauniq (wave): gen firstwave = 1 if obsnr==1 & wave==2
680 * Carry the value of this last variable forwards to the remainder of a participant's observations
681 bysort idauniq: gen firstwave_cons = firstwave[1]
682 * Install unique command
683 ssc install unique
684 * Count total number of participants and observations
685 unique idauniq
686 * 19,807 individuals, 80,750 observations
687 * Save dataset with a new name
688 save raw.dta
689
690 * 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)
691 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)
692 * Count total number of participants and observations
693 unique idauniq
694 * 15,022 individuals, 68,496 observations
695 * Replace age = 90 if participant is aged 90+ years (collapsed in ELSA and coded as 99 at Wave 2, 3,
and 4)
696 replace indager = 90 if indager==99 & inlist(wave,2,3,4)
697 * Replace age = 90 if participant is aged 90+ years (collapsed in ELSA and coded as -7 at Wave 5, 6,
7, 8, and 9)
698 replace indager = 90 if indager==-7 & inlist(wave,5,6,7,8,9)
699 * Save dataset with a new name
700 save data.dta
701
702 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
703 * [b37-b40] Socio-cultural trips (reversed)
704 * [b7-b8] Holidays
705 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
706 * [b37] How often respondent goes to the cinema
707 replace scacta = . if scacta<0
708 * [b38] How often respondent eats out of the house

```

```

709 replace scactb = . if scactb<0
710 * [b39] How often respondent goes to an art gallery or museum
711 replace scactc = . if scactc<0
712 * [b40] How often respondent goes to the theatre, a concert, or the opera
713 replace scactd = . if scactd<0
714 * [b7] Respondent has taken a holiday in the UK in the last 12 months
715 replace scpt04 = . if scpt04<0
716 * [b8] Respondent has taken a holiday abroad in the last 12 months
717 replace scpt05 = . if scpt05<0
718
719 * Reverse the negatively framed variables (this creates new variables and adds the "rev" prefix to
the original variable names)
720 revrs scacta
721 revrs scactb
722 revrs scactc
723 revrs scactd
724 * Collapse categories with a small number of participants
725 replace revscacta = 5 if revscacta == 6
726 replace revscactc = 5 if revscactc == 6
727 replace revscactd = 5 if revscactd == 6
728 replace revscactb = 1 if revscactb ==2
729 replace revscactb = 2 if revscactb ==3
730 replace revscactb = 3 if revscactb ==4
731 replace revscactb = 4 if revscactb ==5
732 replace revscactb = 5 if revscactb ==6
733
734 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
735 * [b26-b36] (+ 2 omitted) ADL and IADL disabilities (reversed)
736 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
737 replace heada01 = . if heada01<0
738 replace heada02 = . if heada02<0
739 replace heada03 = . if heada03<0
740 replace heada04 = . if heada04<0
741 replace heada05 = . if heada05<0
742 replace heada06 = . if heada06<0
743 replace heada07 = . if heada07<0
744 replace heada08 = . if heada08<0
745 replace heada09 = . if heada09<0
746 replace heada10 = . if heada10<0
747 replace headb01 = . if headb01<0
748 replace headb02 = . if headb02<0
749 replace headb03 = . if headb03<0
750 replace headb04 = . if headb04<0
751 replace headb05 = . if headb05<0
752 replace headb06 = . if headb06<0
753 replace headb07 = . if headb07<0
754 replace headb08 = . if headb08<0
755 replace headb09 = . if headb09<0
756 replace headb10 = . if headb10<0
757 replace headb11 = . if headb11<0
758 replace headb12 = . if headb12<0
759 replace headb13 = . if headb13<0
760
761 replace headldr = . if headldr<0
762 replace headlwa = . if headlwa<0
763 replace headlba = . if headlba<0
764 replace headlea = . if headlea<0
765 replace headlbe = . if headlbe<0
766 replace headlwc = . if headlwc<0
767 replace headlma = . if headlma<0
768 replace headlpr = . if headlpr<0
769 replace headlsh = . if headlsh<0
770 replace headlph = . if headlph<0

```



```

771 replace headlme = . if headlme<0
772 replace headlho = . if headlho<0
773 replace headlmo = . if headlmo<0
774 replace headl96 = . if headl96<0
775
776 * ADL
777 * [b26] ADL: difficulty dressing, including putting on shoes and socks
778 replace headldr = 2 if headldr == 0
779 * Assign the number 0 if the participant reported difficulties performing the first listed activity
780 replace headldr = 0 if headldr == 1
781 * Assign the number 1 if the participant reported no difficulties performing the first listed activity
782 replace headldr = 1 if headldr == 2
783 * Assign the number 0 if the participant reported difficulties performing the first listed activity
784 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
785 * Assign the number 1 if the participant reported any answer other than the listed activity in
headb01-headb13 and data are not missing
786 replace headldr = 1 if inlist(headb01,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
787 replace headldr = 1 if inlist(headb02,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
788 replace headldr = 1 if inlist(headb03,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
789 replace headldr = 1 if inlist(headb04,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
790 replace headldr = 1 if inlist(headb05,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
791 replace headldr = 1 if inlist(headb06,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
792 replace headldr = 1 if inlist(headb07,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
793 replace headldr = 1 if inlist(headb08,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
794 replace headldr = 1 if inlist(headb09,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
795 replace headldr = 1 if inlist(headb10,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
796 replace headldr = 1 if inlist(headb11,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
797 replace headldr = 1 if inlist(headb12,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
798 replace headldr = 1 if inlist(headb13,2,3,4,5,6,7,8,9,10,11,12,13,96) & headldr!=0
799
800 * [b27] ADL: difficulty walking across a room
801 replace headlwa = 2 if headlwa == 0
802 * Assign the number 0 if the participant reported difficulties performing the second listed activity
803 replace headlwa = 0 if headlwa == 1
804 * Assign the number 1 if the participant reported no difficulties performing the second listed
activity
805 replace headlwa = 1 if headlwa == 2
806 * Assign the number 0 if the participant reported difficulties performing the second listed activity
807 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
808 * Assign the number 1 if the participant reported no difficulties performing the second listed
activity
809 replace headlwa = 1 if inlist(headb01,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
810 replace headlwa = 1 if inlist(headb02,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
811 replace headlwa = 1 if inlist(headb03,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
812 replace headlwa = 1 if inlist(headb04,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
813 replace headlwa = 1 if inlist(headb05,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
814 replace headlwa = 1 if inlist(headb06,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
815 replace headlwa = 1 if inlist(headb07,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
816 replace headlwa = 1 if inlist(headb08,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
817 replace headlwa = 1 if inlist(headb09,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
818 replace headlwa = 1 if inlist(headb10,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
819 replace headlwa = 1 if inlist(headb11,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
820 replace headlwa = 1 if inlist(headb12,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
821 replace headlwa = 1 if inlist(headb13,1,3,4,5,6,7,8,9,10,11,12,13,96) & headlwa!=0
822
823 * [b28] ADL: difficulty bathing or showering
824 replace headlba = 2 if headlba == 0
825 * Assign the number 0 if the participant reported difficulties performing the third listed activity
826 replace headlba = 0 if headlba == 1

```



```

827 * Assign the number 1 if the participant reported no difficulties performing the third listed activity
828 replace headlba = 1 if headlba == 2
829 * Assign the number 0 if the participant reported difficulties performing the third listed activity
830 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
831 * Assign the number 1 if the participant reported no difficulties performing the third listed activity
832 replace headlba = 1 if inlist(headb01,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
833 replace headlba = 1 if inlist(headb02,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
834 replace headlba = 1 if inlist(headb03,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
835 replace headlba = 1 if inlist(headb04,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
836 replace headlba = 1 if inlist(headb05,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
837 replace headlba = 1 if inlist(headb06,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
838 replace headlba = 1 if inlist(headb07,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
839 replace headlba = 1 if inlist(headb08,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
840 replace headlba = 1 if inlist(headb09,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
841 replace headlba = 1 if inlist(headb10,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
842 replace headlba = 1 if inlist(headb11,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
843 replace headlba = 1 if inlist(headb12,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
844 replace headlba = 1 if inlist(headb13,1,2,4,5,6,7,8,9,10,11,12,13,96) & headlba!=0
845
846 * [b29] ADL: difficulty eating, such as cutting up your food
847 replace headlea = 2 if headlea == 0
848 * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
849 replace headlea = 0 if headlea == 1
850 * Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity
851 replace headlea = 1 if headlea == 2
852 * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
853 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
854 * Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity
855 replace headlea = 1 if inlist(headb01,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
856 replace headlea = 1 if inlist(headb02,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
857 replace headlea = 1 if inlist(headb03,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
858 replace headlea = 1 if inlist(headb04,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
859 replace headlea = 1 if inlist(headb05,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
860 replace headlea = 1 if inlist(headb06,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
861 replace headlea = 1 if inlist(headb07,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
862 replace headlea = 1 if inlist(headb08,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
863 replace headlea = 1 if inlist(headb09,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
864 replace headlea = 1 if inlist(headb10,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
865 replace headlea = 1 if inlist(headb11,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
866 replace headlea = 1 if inlist(headb12,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
867 replace headlea = 1 if inlist(headb13,1,2,3,5,6,7,8,9,10,11,12,13,96) & headlea!=0
868
869 * [b30] ADL: difficulty getting in or out of bed
870 replace headlbe = 2 if headlbe == 0
871 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
872 replace headlbe = 0 if headlbe == 1
873 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
874 replace headlbe = 1 if headlbe == 2
875 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
876 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
877 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
878 replace headlbe = 1 if inlist(headb01,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
879 replace headlbe = 1 if inlist(headb02,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
880 replace headlbe = 1 if inlist(headb03,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
881 replace headlbe = 1 if inlist(headb04,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0

```

```

882 replace headlbe = 1 if inlist(headb05,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
883 replace headlbe = 1 if inlist(headb06,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
884 replace headlbe = 1 if inlist(headb07,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
885 replace headlbe = 1 if inlist(headb08,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
886 replace headlbe = 1 if inlist(headb09,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
887 replace headlbe = 1 if inlist(headb10,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
888 replace headlbe = 1 if inlist(headb11,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
889 replace headlbe = 1 if inlist(headb12,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
890 replace headlbe = 1 if inlist(headb13,1,2,3,4,6,7,8,9,10,11,12,13,96) & headlbe!=0
891
892 * [b31] ADL: difficulty using the toilet, including getting up or down
893 replace headlwc = 2 if headlwc == 0
894 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
895 replace headlwc = 0 if headlwc == 1
896 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
897 replace headlwc = 1 if headlwc == 2
898 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
899 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
900 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
901 replace headlwc = 1 if inlist(headb01,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
902 replace headlwc = 1 if inlist(headb02,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
903 replace headlwc = 1 if inlist(headb03,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
904 replace headlwc = 1 if inlist(headb04,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
905 replace headlwc = 1 if inlist(headb05,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
906 replace headlwc = 1 if inlist(headb06,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
907 replace headlwc = 1 if inlist(headb07,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
908 replace headlwc = 1 if inlist(headb08,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
909 replace headlwc = 1 if inlist(headb09,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
910 replace headlwc = 1 if inlist(headb10,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
911 replace headlwc = 1 if inlist(headb11,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
912 replace headlwc = 1 if inlist(headb12,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
913 replace headlwc = 1 if inlist(headb13,1,2,3,4,5,7,8,9,10,11,12,13,96) & headlwc!=0
914
915 * IADL
916 * [b32] IADL: difficulty using a map to figure out how to get around in a strange place
917 replace headlma = 2 if headlma == 0
918 * Assign the number 0 if the participant reported difficulties performing the first listed activity
919 replace headlma = 0 if headlma == 1
920 * Assign the number 1 if the participant reported no difficulties performing the first listed activity
921 replace headlma = 1 if headlma == 2
922 * Assign the number 0 if the participant reported difficulties performing the first listed activity
923 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
924 * Assign the number 1 if the participant reported no difficulties performing the first listed activity
925 replace headlma = 1 if inlist(headb01,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
926 replace headlma = 1 if inlist(headb02,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
927 replace headlma = 1 if inlist(headb03,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
928 replace headlma = 1 if inlist(headb04,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
929 replace headlma = 1 if inlist(headb05,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
930 replace headlma = 1 if inlist(headb06,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
931 replace headlma = 1 if inlist(headb07,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
932 replace headlma = 1 if inlist(headb08,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
933 replace headlma = 1 if inlist(headb09,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
934 replace headlma = 1 if inlist(headb10,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
935 replace headlma = 1 if inlist(headb11,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
936 replace headlma = 1 if inlist(headb12,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
937 replace headlma = 1 if inlist(headb13,1,2,3,4,5,6,8,9,10,11,12,13,96) & headlma!=0
938
939 * [b33] IADL: difficulty preparing a hot meal
940 replace headlpr = 2 if headlpr == 0

```

```

941 * Assign the number 0 if the participant reported difficulties performing the second listed activity
942 replace headlpr = 0 if headlpr == 1
943 * Assign the number 1 if the participant reported no difficulties performing the second listed
activity
944 replace headlpr = 1 if headlpr == 2
945 * Assign the number 0 if the participant reported difficulties performing the second listed activity
946 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
947 * Assign the number 1 if the participant reported no difficulties performing the second listed
activity
948 replace headlpr = 1 if inlist(headb01,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
949 replace headlpr = 1 if inlist(headb02,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
950 replace headlpr = 1 if inlist(headb03,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
951 replace headlpr = 1 if inlist(headb04,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
952 replace headlpr = 1 if inlist(headb05,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
953 replace headlpr = 1 if inlist(headb06,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
954 replace headlpr = 1 if inlist(headb07,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
955 replace headlpr = 1 if inlist(headb08,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
956 replace headlpr = 1 if inlist(headb09,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
957 replace headlpr = 1 if inlist(headb10,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
958 replace headlpr = 1 if inlist(headb11,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
959 replace headlpr = 1 if inlist(headb12,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
960 replace headlpr = 1 if inlist(headb13,1,2,3,4,5,6,7,9,10,11,12,13,96) & headlpr!=0
961
962 * [b34] IADL: difficulty shopping for groceries
963 replace headlsh = 2 if headlsh == 0
964 * Assign the number 0 if the participant reported difficulties performing the third listed activity
965 replace headlsh = 0 if headlsh == 1
966 * Assign the number 1 if the participant reported no difficulties performing the third listed activity
967 replace headlsh = 1 if headlsh == 2
968 * Assign the number 0 if the participant reported difficulties performing the third listed activity
969 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
970 * Assign the number 1 if the participant reported no difficulties performing the third listed activity
971 replace headlsh = 1 if inlist(headb01,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
972 replace headlsh = 1 if inlist(headb02,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
973 replace headlsh = 1 if inlist(headb03,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
974 replace headlsh = 1 if inlist(headb04,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
975 replace headlsh = 1 if inlist(headb05,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
976 replace headlsh = 1 if inlist(headb06,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
977 replace headlsh = 1 if inlist(headb07,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
978 replace headlsh = 1 if inlist(headb08,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
979 replace headlsh = 1 if inlist(headb09,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
980 replace headlsh = 1 if inlist(headb10,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
981 replace headlsh = 1 if inlist(headb11,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
982 replace headlsh = 1 if inlist(headb12,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
983 replace headlsh = 1 if inlist(headb13,1,2,3,4,5,6,7,8,10,11,12,13,96) & headlsh!=0
984
985 * [b35] IADL: difficulty making telephone calls
986 replace headlph = 2 if headlph == 0
987 * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
988 replace headlph = 0 if headlph == 1
989 * Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity
990 replace headlph = 1 if headlph == 2
991 * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
992 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
993 * Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity

```

```

994 replace headlph = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
995 replace headlph = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
996 replace headlph = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
997 replace headlph = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
998 replace headlph = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
999 replace headlph = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1000 replace headlph = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1001 replace headlph = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1002 replace headlph = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1003 replace headlph = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1004 replace headlph = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1005 replace headlph = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1006 replace headlph = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,11,12,13,96) & headlph!=0
1007
1008 * (1 omitted) IADL: difficulty talking medications
1009 replace headlme = 2 if headlme == 0
1010 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
1011 replace headlme = 0 if headlme == 1
1012 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
1013 replace headlme = 1 if headlme == 2
1014 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
1015 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
1016 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
1017 replace headlme = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1018 replace headlme = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1019 replace headlme = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1020 replace headlme = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1021 replace headlme = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1022 replace headlme = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1023 replace headlme = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1024 replace headlme = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1025 replace headlme = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1026 replace headlme = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1027 replace headlme = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1028 replace headlme = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1029 replace headlme = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,12,13,96) & headlme!=0
1030
1031 * [b36] IADL: difficulty doing work around the house or garden
1032 replace headlho = 2 if headlho == 0
1033 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
1034 replace headlho = 0 if headlho == 1
1035 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
1036 replace headlho = 1 if headlho == 2
1037 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
1038 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
1039 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
1040 replace headlho = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1041 replace headlho = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1042 replace headlho = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1043 replace headlho = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1044 replace headlho = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1045 replace headlho = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1046 replace headlho = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1047 replace headlho = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1048 replace headlho = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1049 replace headlho = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1050 replace headlho = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1051 replace headlho = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0
1052 replace headlho = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,11,13,96) & headlho!=0

```



```

1053
1054 * (1 omitted) IADL: difficulty managing money, such as paying bills and keeping track of expenses
1055 replace headlmo = 2 if headlmo == 0
1056 * Assign the number 0 if the participant reported difficulties performing the seventh listed activity
1057 replace headlmo = 0 if headlmo == 1
1058 * Assign the number 1 if the participant reported no difficulties performing the seventh listed
activity
1059 replace headlmo = 1 if headlmo == 2
1060 * Assign the number 0 if the participant reported difficulties performing the seventh listed activity
1061 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
1062 * Assign the number 1 if the participant reported no difficulties performing the seventh listed
activity
1063 replace headlmo = 1 if inlist(headb01,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1064 replace headlmo = 1 if inlist(headb02,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1065 replace headlmo = 1 if inlist(headb03,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1066 replace headlmo = 1 if inlist(headb04,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1067 replace headlmo = 1 if inlist(headb05,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1068 replace headlmo = 1 if inlist(headb06,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1069 replace headlmo = 1 if inlist(headb07,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1070 replace headlmo = 1 if inlist(headb08,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1071 replace headlmo = 1 if inlist(headb09,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1072 replace headlmo = 1 if inlist(headb10,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1073 replace headlmo = 1 if inlist(headb11,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1074 replace headlmo = 1 if inlist(headb12,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1075 replace headlmo = 1 if inlist(headb13,1,2,3,4,5,6,7,8,9,10,11,12,96) & headlmo!=0
1076
1077 * Save dataset with a new name
1078 save datavariables.dta
1079
1080 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1081 * [b16-b25] Mobility (reversed)
1082 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1083 replace hemobwa = . if hemobwa<0
1084 replace hemobsi = . if hemobsi<0
1085 replace hemobch = . if hemobch<0
1086 replace hemobcs = . if hemobcs<0
1087 replace hemobcl = . if hemobcl<0
1088 replace hemobst = . if hemobst<0
1089 replace hemobre = . if hemobre<0
1090 replace hemobpu = . if hemobpu<0
1091 replace hemobli = . if hemobli<0
1092 replace hemobpi = . if hemobpi<0
1093 replace hemob96 = . if hemob96<0
1094
1095 * [b16] Mobility: difficulty walking 100 yards
1096 replace hemobwa = 2 if hemobwa == 0
1097 * Assign the number 0 if the participant reported difficulties performing the first listed activity
1098 replace hemobwa = 0 if hemobwa == 1
1099 * Assign the number 1 if the participant reported no difficulties performing the first listed activity
1100 replace hemobwa = 1 if hemobwa == 2
1101 * Assign the number 0 if the participant reported difficulties performing the first listed activity
1102 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
1103 * Assign the number 1 if the participant reported no difficulties performing the first listed activity
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 no difficulties performing the second listed
activity
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 no difficulties performing the third listed activity
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
1160 replace hemobcs = 1 if hemobcs == 2
1161 * Assign the number 0 if the participant reported difficulties performing the fourth listed activity
1162 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
1163 * Assign the number 1 if the participant reported no difficulties performing the fourth listed
activity
1164 replace hemobcs = 1 if inlist(heada01,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1165 replace hemobcs = 1 if inlist(heada02,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1166 replace hemobcs = 1 if inlist(heada03,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0

```

```

1167 replace hemobcs = 1 if inlist(heada04,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1168 replace hemobcs = 1 if inlist(heada05,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1169 replace hemobcs = 1 if inlist(heada06,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1170 replace hemobcs = 1 if inlist(heada07,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1171 replace hemobcs = 1 if inlist(heada08,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1172 replace hemobcs = 1 if inlist(heada09,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1173 replace hemobcs = 1 if inlist(heada10,1,2,3,5,6,7,8,9,10,96) & hemobcs!=0
1174
1175 * [b20] Mobility: difficulty climbing one flight of stairs without resting
1176 replace hemobcl = 2 if hemobcl == 0
1177 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
1178 replace hemobcl = 0 if hemobcl == 1
1179 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
1180 replace hemobcl = 1 if hemobcl == 2
1181 * Assign the number 0 if the participant reported difficulties performing the fifth listed activity
1182 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
1183 * Assign the number 1 if the participant reported no difficulties performing the fifth listed activity
1184 replace hemobcl = 1 if inlist(heada01,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1185 replace hemobcl = 1 if inlist(heada02,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1186 replace hemobcl = 1 if inlist(heada03,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1187 replace hemobcl = 1 if inlist(heada04,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1188 replace hemobcl = 1 if inlist(heada05,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1189 replace hemobcl = 1 if inlist(heada06,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1190 replace hemobcl = 1 if inlist(heada07,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1191 replace hemobcl = 1 if inlist(heada08,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1192 replace hemobcl = 1 if inlist(heada09,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1193 replace hemobcl = 1 if inlist(heada10,1,2,3,4,6,7,8,9,10,96) & hemobcl!=0
1194
1195 * [b21] Mobility: difficulty stooping, kneeling, or crouching
1196 replace hemobst = 2 if hemobst == 0
1197 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
1198 replace hemobst = 0 if hemobst == 1
1199 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
1200 replace hemobst = 1 if hemobst == 2
1201 * Assign the number 0 if the participant reported difficulties performing the sixth listed activity
1202 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
1203 * Assign the number 1 if the participant reported no difficulties performing the sixth listed activity
1204 replace hemobst = 1 if inlist(heada01,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1205 replace hemobst = 1 if inlist(heada02,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1206 replace hemobst = 1 if inlist(heada03,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1207 replace hemobst = 1 if inlist(heada04,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1208 replace hemobst = 1 if inlist(heada05,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1209 replace hemobst = 1 if inlist(heada06,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1210 replace hemobst = 1 if inlist(heada07,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1211 replace hemobst = 1 if inlist(heada08,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1212 replace hemobst = 1 if inlist(heada09,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1213 replace hemobst = 1 if inlist(heada10,1,2,3,4,5,7,8,9,10,96) & hemobst!=0
1214
1215 * [b22] Mobility: difficulty reaching or extending arms above shoulder level
1216 replace hemobre = 2 if hemobre == 0
1217 * Assign the number 0 if the participant reported difficulties performing the seventh listed activity
1218 replace hemobre = 0 if hemobre == 1
1219 * Assign the number 1 if the participant reported no difficulties performing the seventh listed
activity
1220 replace hemobre = 1 if hemobre == 2
1221 * Assign the number 0 if the participant reported difficulties performing the seventh listed activity
1222 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
1223 * Assign the number 1 if the participant reported no difficulties performing the seventh listed
activity
1224 replace hemobre = 1 if inlist(heada01,1,2,3,4,5,6,8,9,10,96) & hemobre!=0

```



```

1225 replace hemobre = 1 if inlist(heada02,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1226 replace hemobre = 1 if inlist(heada03,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1227 replace hemobre = 1 if inlist(heada04,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1228 replace hemobre = 1 if inlist(heada05,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1229 replace hemobre = 1 if inlist(heada06,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1230 replace hemobre = 1 if inlist(heada07,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1231 replace hemobre = 1 if inlist(heada08,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1232 replace hemobre = 1 if inlist(heada09,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1233 replace hemobre = 1 if inlist(heada10,1,2,3,4,5,6,8,9,10,96) & hemobre!=0
1234
1235 * [b23] Mobility: difficulty pulling or pushing large objects, like a living room chair
1236 replace hemobpu = 2 if hemobpu == 0
1237 * Assign the number 0 if the participant reported difficulties performing the eighth listed activity
1238 replace hemobpu = 0 if hemobpu == 1
1239 * Assign the number 1 if the participant reported no difficulties performing the eighth listed
activity
1240 replace hemobpu = 1 if hemobpu == 2
1241 * Assign the number 0 if the participant reported difficulties performing the eighth listed activity
1242 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
1243 * Assign the number 1 if the participant reported no difficulties performing the eighth listed
activity
1244 replace hemobpu = 1 if inlist(heada01,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1245 replace hemobpu = 1 if inlist(heada02,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1246 replace hemobpu = 1 if inlist(heada03,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1247 replace hemobpu = 1 if inlist(heada04,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1248 replace hemobpu = 1 if inlist(heada05,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1249 replace hemobpu = 1 if inlist(heada06,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1250 replace hemobpu = 1 if inlist(heada07,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1251 replace hemobpu = 1 if inlist(heada08,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1252 replace hemobpu = 1 if inlist(heada09,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1253 replace hemobpu = 1 if inlist(heada10,1,2,3,4,5,6,7,9,10,96) & hemobpu!=0
1254
1255 * [b24] Mobility: difficulty lifting or carrying weights over 10 pounds, like a heavy bag of groceries
1256 replace hemobli = 2 if hemobli == 0
1257 * Assign the number 0 if the participant reported difficulties performing the ninth listed activity
1258 replace hemobli = 0 if hemobli == 1
1259 * Assign the number 1 if the participant reported no difficulties performing the ninth listed activity
1260 replace hemobli = 1 if hemobli == 2
1261 * Assign the number 0 if the participant reported difficulties performing the ninth listed activity
1262 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
1263 * Assign the number 1 if the participant reported no difficulties performing the ninth listed activity
1264 replace hemobli = 1 if inlist(heada01,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1265 replace hemobli = 1 if inlist(heada02,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1266 replace hemobli = 1 if inlist(heada03,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1267 replace hemobli = 1 if inlist(heada04,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1268 replace hemobli = 1 if inlist(heada05,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1269 replace hemobli = 1 if inlist(heada06,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1270 replace hemobli = 1 if inlist(heada07,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1271 replace hemobli = 1 if inlist(heada08,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1272 replace hemobli = 1 if inlist(heada09,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1273 replace hemobli = 1 if inlist(heada10,1,2,3,4,5,6,7,8,10,96) & hemobli!=0
1274
1275 * [b25] Mobility: difficulty picking up a 5p coin from a table
1276 replace hemobpi = 2 if hemobpi == 0
1277 * Assign the number 0 if the participant reported difficulties performing the tenth listed activity
1278 replace hemobpi = 0 if hemobpi == 1
1279 * Assign the number 1 if the participant reported no difficulties performing the tenth listed activity
1280 replace hemobpi = 1 if hemobpi == 2
1281 * Assign the number 0 if the participant reported difficulties performing the tenth listed activity
1282 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

```

```

1283 * Assign the number 1 if the participant reported no difficulties performing the tenth listed activity
1284 replace hemobpi = 1 if inlist(heada01,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1285 replace hemobpi = 1 if inlist(heada02,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1286 replace hemobpi = 1 if inlist(heada03,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1287 replace hemobpi = 1 if inlist(heada04,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1288 replace hemobpi = 1 if inlist(heada05,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1289 replace hemobpi = 1 if inlist(heada06,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1290 replace hemobpi = 1 if inlist(heada07,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1291 replace hemobpi = 1 if inlist(heada08,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1292 replace hemobpi = 1 if inlist(heada09,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1293 replace hemobpi = 1 if inlist(heada10,1,2,3,4,5,6,7,8,9,96) & hemobpi!=0
1294
1295 * Overwrite dataset, by replacing the previously saved file
1296 save datavariables.dta, replace
1297
1298 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1299 * Socio-economic covariate - Quintiles of BU total (non-pension) wealth
1300 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1301 replace totwq5_bu_s = . if totwq5_bu_s<0
1302
1303 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1304 * [b41] Whether has self-reported limiting long-standing illness (reversed)
1305 * Generate a new variable and assign the number 0 for participants with a limiting long-standing
    illness
1306 gen limiting = 0 if helim==1
1307 * Assign the number 1 for participants with no long-standing illness or a long-standing illness that
    is not limiting
1308 replace limiting = 1 if heill == 2 | helim == 2
1309
1310 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1311 * [b9-b15] (+ 1 omitted) Organisational memberships
1312 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1313 * [b9] Political party, trade union or environmental group
1314 replace scorg01 = . if scorg01<0
1315 * [b10] Tenants or resident group or neighbourhood watch
1316 replace scorg02 = . if scorg02<0
1317 * [b11] Member of a church or other religious group
1318 replace scorg03 = . if scorg03<0
1319 * [b12] Member of a charitable association
1320 replace scorg04 = . if scorg04<0
1321 * [b13] An education, arts or music group or evening class
1322 replace scorg05 = . if scorg05<0
1323 * (1 omitted) Member of a social club
1324 replace scorg06 = . if scorg06<0
1325 * [b14] Member of a sports club, gym, or exercise class
1326 replace scorg07 = . if scorg07<0
1327 * [b15] Member of any other organisations, clubs, or societies
1328 replace scorg08 = . if scorg08<0
1329
1330 * Save dataset with a new name
1331 save datavariables01.dta
1332
1333 * [b42-b43] (+ 1 omitted) Short Physical Performance Battery
1334 * WAVE 2, 4, 6
1335 * (1 omitted) Standing balance
1336 tab mmbcsc
1337 tab mmsssc
1338 tab mmsre
1339 tab mmsssc if mmsre==3
1340 tab mmbcsc if mmsssc<0
1341 tab mmsssc if mmsre<0
1342 sum mmssti if mmssti!=-1
1343 tab mmsre if mmssti!=-1

```

```

1344 sum mmssna if mmssna!=-1
1345 tab mmssna if mmssna!=-1 & wave==2
1346 tab mmssna if mmssna!=-1 & inlist(wave,4,6)
1347 tab mmsssc if mmssna!=-1
1348
1349 tab mmstsc
1350 tab mmstsc if mmssna!=-1
1351 tab mmssre if mmstsc==1
1352 tab mmstre
1353 tab mmstsc if mmstre==3
1354 sum mmstti if mmstti!=-1
1355 tab mmstre if mmstti!=-1
1356 sum mmstna if mmstna!=-1
1357 tab mmstna if mmstna!=-1 & wave==2
1358 tab mmstna if mmstna!=-1 & inlist(wave,4,6)
1359 tab mmstsc if mmstna!=-1
1360
1361 tab mmftsc
1362 tab mmftsc if mmstna!=-1
1363 tab mmftsc if mmstti!=-1
1364 tab mmftsc if mmssna!=-1 | mmssti!=-1
1365 tab mmftsc if mmsssc<0
1366 tab mmftre2
1367 tab mmftsc if mmftre2==5
1368 sum mmftti if mmftti!=-1
1369 sum mmftti if inlist(mmftre2,2,4)
1370 tab mmftre2 if mmftti!=-1
1371 tab mmftre2 if inlist(mmftre2,2,4)
1372 sum mmftti if mmftti >=3 & mmftti < 10
1373 sum mmftti if mmftti < 3 & mmftti!=-1
1374 tab mmftre2 if mmftti >=10 & mmftti!=.
1375 sum mmftti if mmftti >=10 & mmftti!=.
1376 sum mmftna if mmftna!=-1
1377 tab mmftna if mmftna!=-1 & wave==2
1378 tab mmftna if mmftna!=-1 & inlist(wave,4,6)
1379 tab mmftsc if mmftna!=-1
1380 tab mmftre2 if wave==6
1381 tab mmftre
1382
1383 * Side-by-side stand
1384 * Generate a new variable and assign the number 0 if the participant held for less than 10 seconds
or did not attempt the stand
1385 gen sidebyside = 0 if inlist(mmsre,2,3)
1386 * Assign the number 1 if the participant held for 10 seconds
1387 replace sidebyside = 1 if mmsre==1
1388
1389 * Semi-tandem stand
1390 * Generate a new variable and assign the number 0 if the participant scored 0 points in the
side-by-side stand
1391 gen semitandem = 0 if sidebyside==0
1392 * Assign the number 0 if the participant held for less than 10 seconds or did not attempt the stand
1393 replace semitandem = 0 if inlist(mmstre,2,3)
1394 * Assign the number 1 if the participant held for 10 seconds
1395 replace semitandem = 1 if mmstre==1
1396
1397 * Full tandem stand
1398 * 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
1399 gen tandem = 0 if (mmftre2 == 5 | semitandem == 0)
1400 * Assign the number 2 if the participant held for at least 10 seconds, irrespective of age
1401 replace tandem = 2 if (mmftre2 == 1 | mmftre2 == 3)
1402 * Assign the number 1 if the participant held for at least 3 seconds but less than 10 seconds
1403 replace tandem = 1 if mmftti >=3 & mmftti < 10

```

```

1404 * Assign the number 0 if the participant held for less than 3 seconds and data are not missing
1405 replace tandem = 0 if mmftti < 3 & mmftti!=.-1
1406 * 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
1407 replace tandem = 2 if mmftti >=10 & mmftti!=. & mmftre2==4
1408
1409 * 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)
1410 gen balance = sidebyside + semitandem + tandem
1411
1412 gen balance2 = 0 if inlist(mssre,2,3)
1413 replace balance2 = 1 if mssre==1 & inlist(mmstre,2,3)
1414 replace balance2 = 2 if mmstre==1 & tandem==0
1415 replace balance2 = 3 if mmstre==1 & tandem==1
1416 replace balance2 = 4 if mmstre==1 & tandem==2
1417
1418 * WAVE 2, 4, 6
1419 * [b42] Repeated chair stands
1420 tab mmcrav
1421 * Generate a new variable and assign a missing value if there was no suitable chair available or
data are missing
1422 gen repctest = . if mmcrav==2 | mmcrav<0
1423 tab mmcrsc
1424 tab mmcrsc if mmcrav==2 | mmcrav<0
1425 tab mmcrre
1426 tab mmcrre if mmcrav==2 | mmcrav<0 | inlist(mmcrsc,-8,2)
1427 * 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"
1428 replace repctest = 0 if inlist(mmcrsc,-8,2)
1429 * Assign the number 0 if the participant used their arms to stand in the single chair rise or did
not complete the test
1430 replace repctest = 0 if inlist(mmcrre,2,3)
1431 tab mmcrav if mmcrre==.-1
1432 tab mmcrsc if mmcrre==.-1
1433 sum mmcrna if mmcrna!=.-1
1434 tab mmcrna if mmcrna!=.-1 & wave==2
1435 tab mmcrna if mmcrna!=.-1 & inlist(wave,4,6)
1436 tab mmrrsc
1437 tab mmcrre if mmrrsc==.-1
1438 tab mmrrre
1439 tab mmrrsc if mmrrre==.-1
1440 tab mmrrre if inlist(mmrrsc,2,-1)
1441 * 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)
1442 replace repctest = 0 if mmrrsc==2 & mmrrre==.-1
1443 * Assign the number 0 if the participant completed less than five sit-to-stands
1444 replace repctest = 0 if inlist(mmrrre,0,1,2,3,4)
1445 tab mmrrfti if mmrrfti<0
1446 tab mmrrre if mmrrfti<0
1447 sum mmrrfti if mmrrre>=5
1448 sum mmrrfti if mmrrfti>=0
1449 tab mmrrfti if mmrrfti<0 & wave==2
1450 tab mmrrfti if mmrrfti<0 & wave==4
1451 tab mmrrfti if mmrrfti<0 & wave==6
1452 * 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
1453 replace repctest = . if inlist(mmrrfti,-8,-3) & mmrrre>=5
1454 tab mmrrfti if mmrrfti>=0 & mmrrfti<4
1455 * 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)
1456 replace repctest = . if inlist(mmrrfti,0,1) & inlist(mmrrre,5,6,7,8,9,10)
1457 sum mmrrfti if mmrrfti<=11.19 & mmrrfti>=0
1458 sum mmrrfti if mmrrfti<=11.19 & mmrrfti>1

```

```

1459 * Assign the number 4 if the participant completed five sit-to-stands in less than or equal to 11.19
1460 seconds and it took them more than 1 second
1461 replace repctest = 4 if mmrrfti <= 11.19 & mmrrfti > 1 & inlist(mmrrre,5,6,7,8,9,10)
1462 sum mmrrfti if mmrrfti>=16.7 & mmrrfti<=60
1463 * Assign the number 1 if the participant completed five sit-to-stands in 16.7 seconds or more but
1464 less than 60 seconds
1465 replace repctest = 1 if mmrrfti >= 16.7 & mmrrfti <= 60 & inlist(mmrrre,5,6,7,8,9,10)
1466 sum mmrrfti if mmrrfti>=13.7 & mmrrfti<16.7
1467 * Assign the number 2 if the participant completed five sit-to-stands in 13.7 seconds or more but
1468 less than 16.7 seconds
1469 replace repctest = 2 if mmrrfti >= 13.7 & mmrrfti < 16.7 & inlist(mmrrre,5,6,7,8,9,10)
1470 sum mmrrfti if mmrrfti>=11.2 & mmrrfti<13.7
1471 * Assign the number 3 if the participant completed five sit-to-stands in 11.2 seconds or more but
1472 less than 13.7 seconds
1473 replace repctest = 3 if mmrrfti >= 11.2 & mmrrfti < 13.7 & inlist(mmrrre,5,6,7,8,9,10)
1474 sum mmrrfti if mmrrfti > 60
1475 * Assign the number 0 if the participant completed five sit-to-stands in more than 60 seconds and
1476 data are not missing
1477 replace repctest = 0 if mmrrfti > 60 & mmrrfti <= 64 & inlist(mmrrre,5,6,7,8,9,10)
1478
1479 tab mmrrre if inlist(mmrrroc,1,3)
1480 tab mmrrre if mmrrroc==2
1481 tab mmrrre if mmrrroc==4
1482 tab mmrrre if inlist(mmrrroc,1,2,3,4)
1483
1484 sum mmrrna if mmrrna!=-1
1485 tab mmrrna if mmrrna!=-1 & wave==2
1486 tab mmrrna if mmrrna!=-1 & inlist(wave,4,6)
1487 tab mmrrre if mmrrna!=-1 & inlist(mmrrroc,3,4)
1488 tab mmrrre if mmrrna!=-1 & inlist(mmrrroc,1,2)
1489 tab mmrrsc if mmrrna!=-1
1490 tab mmrrna if mmrrre==5 & inlist(mmrrroc,1,2)
1491 sum mmrrfti if mmrrre==5 & inlist(mmrrroc,1,2) & mmrrna!=-1
1492
1493 * Save dataset with a new name
1494 save datavariabes02.dta
1495
1496 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1497 * [b43] Gait speed
1498 tab mmschs if indager >=60
1499 tab mmalone if indager >=60
1500 tab mmschs if mmalone== -1 & indager >=60
1501 tab mmschs if mmalone== -9 & indager >=60
1502 tab mmschs if mmalone== -8 & indager >=60
1503 tab mmschs if mmalone== -2 & indager >=60
1504 tab mmschs if mmalone== 1 & indager >=60
1505 tab mmschs if mmalone== 2 & indager >=60
1506 tab mmschs if mmalone== 3 & indager >=60
1507 tab mmhss if indager >=60
1508 tab mmalone if mmhss== -1 & indager >=60
1509 tab mmalone if mmhss== -9 & indager >=60
1510 tab mmalone if mmhss== -8 & indager >=60
1511 tab mmalone if mmhss== -2 & indager >=60
1512 tab mmalone if mmhss== 1 & indager >=60
1513 tab mmalone if mmhss== 2 & indager >=60
1514 tab mmalone if mmhss== 3 & indager >=60
1515 tab mmalone if mmhss== 4 & indager >=60
1516 tab mmwill if indager >=60
1517 tab mmhss if mmwill== -9 & indager >=60
1518 tab mmhss if mmwill== -8 & indager >=60
1519 tab mmhss if mmwill== -2 & indager >=60
1520 tab mmhss if mmwill== -1 & indager >=60
1521 tab mmhss if mmwill== 1 & indager >=60

```



```

1517 tab mmhss if mmwill==2 & indager >=60
1518 tab mmsaf if indager >=60
1519 tab mmwill if mmsaf==1 & indager >=60
1520 tab mmwill if mmsaf==2 & indager >=60
1521 tab mmavsp if indager >=60
1522 tab mmsaf if mmavsp== -9 & indager >=60
1523 tab mmsaf if mmavsp== -8 & indager >=60
1524 tab mmsaf if mmavsp== -2 & indager >=60
1525 tab mmsaf if mmavsp== -1 & indager >=60
1526 tab mmsaf if mmavsp== 1 & indager >=60
1527 tab mmsaf if mmavsp== 2 & indager >=60
1528 tab mmwala if indager >=60
1529 tab mmavsp if inlist(mmwala,1,2) & indager >=60
1530
1531 tab mmtrya if indager >=60
1532 tab mmwala if inlist(mmtrya,-9,1,2,3) & indager >=60
1533 tab mmwala if mmtrya==4 & indager >=60
1534 tab mmwala if mmtrya== -8 & indager >=60
1535 tab mmwala if mmtrya== -2 & indager >=60
1536 tab mmwlka if mmwlka<0 & indager >=60
1537 sum mmwlka if mmwlka>=0 & indager >=60
1538 tab mmtrya if mmwlka== -1 & indager >=60
1539 tab mmtrya if mmwlka== -2 & indager >=60
1540 tab mmtrya if mmwlka== -8 & indager >=60
1541 tab mmtrya if mmwlka== -9 & indager >=60
1542 tab mmtrya if mmwlka>=0 & indager >=60
1543
1544 tab mmtryb if indager >=60
1545 tab mmtrya if inlist(mmtryb,-8,1,2,3,4) & indager >=60
1546 tab mmwlka if inlist(mmtryb,-9,-8,-2,-1) & indager >=60
1547 tab mmwlkb if mmwlkb<0 & indager >=60
1548 sum mmwlkb if mmwlkb>=0 & indager >=60
1549 tab mmtryb if mmwlkb== -1 & indager >=60
1550 tab mmtryb if mmwlkb== -2 & indager >=60
1551 tab mmtryb if mmwlkb== -8 & indager >=60
1552 tab mmtryb if mmwlkb== -9 & indager >=60
1553 tab mmtryb if mmwlkb>=0 & indager >=60
1554
1555 tab mmwlka if mmwlka>=0 & mmwlka<2 & indager >=60
1556 tab mmwlkb if mmwlkb>=0 & mmwlkb<2 & indager >=60
1557
1558 sum idauniq if ((mmwlka>=0 & mmwlka!=.) | (mmwlkb>=0 & mmwlkb!=.)) & indager >=60
1559 sum idauniq if mmwlka>=0 & mmwlkb>=0 & mmwlka!=. & mmwlkb!=. & indager >=60
1560
1561 sum idauniq if mmwlka<0 & indager >=60
1562 keep if mmwlka<0 & indager >=60
1563 tab mmschs if indager >=60
1564 tab mmalone if indager >=60
1565 tab mmschs if mmalone== -1 & indager >=60
1566 tab mmschs if mmalone== 3 & indager >=60
1567 tab mmschs if mmalone== -9 & indager >=60
1568 tab mmschs if mmalone== -8 & indager >=60
1569 tab mmschs if mmalone== -2 & indager >=60
1570 tab mmhss if indager >=60
1571 tab mmalone if mmhss== -1 & indager >=60
1572 tab mmwill if indager >=60
1573 tab mmhss if mmwill== -1 & indager >=60
1574 tab mmsaf if indager >=60
1575 tab mmwill if mmsaf== -1 & indager >=60
1576 tab mmavsp if indager >=60
1577 tab mmsaf if mmavsp== -1 & indager >=60
1578 tab mmwala if indager >=60
1579 tab mmavsp if mmwala== -1 & indager >=60

```

```

1580 tab mmtrya if indager >=60
1581 tab mmwala if mmtrya==1 & indager >=60
1582 tab mmwlka if indager >=60
1583 tab mmtrya if mmwlka==1 & indager >=60
1584 tab mmtrya if mmwlka<0 & mmwlka!=1 & indager >=60
1585
1586 clear
1587 * Use full dataset
1588 use datavariab02.dta
1589
1590 * Generate a new variable duplicating the "time taken for first walk" variable if data are not missing
1591 gen walk1 = mmwlka if mmwlka>=0 & indager >=60
1592 * Generate a new variable duplicating the "time taken for second walk" variable if data are not
missing
1593 gen walk2 = mmwlkb if mmwlkb>=0 & indager >=60
1594 * 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)
1595 egen gaittime = rowmin(walk1 walk2) if ((mmwlka>=0 & mmwlka!=.) | (mmwlkb>=0 & mmwlkb!=.)) & indager
>=60
1596 sum gaittime
1597 sum gaittime if (mmwlka>=0 & mmwlka!=.) & (mmwlkb<0|mmwlkb==.)
1598 sum mmwlka if (mmwlka>=0 & mmwlka!=.) & (mmwlkb<0|mmwlkb==.)
1599
1600 * Generate a new variable
1601 gen gaittest = .
1602 * Assign the number 1 if the participant completed the gait test in more than or equal to 5.7 seconds
1603 replace gaittest = 1 if gaittime >= 5.7 & gaittime!=.
1604 * 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
1605 replace gaittest = 2 if gaittime >= 4.1 & gaittime < 5.7
1606 * 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
1607 replace gaittest = 3 if gaittime >= 3.2 & gaittime < 4.1
1608 * Assign the number 4 if the participant completed the gait test in less than 3.2 seconds
1609 replace guralnik = 4 if gaittime < 3.2
1610 * 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
1611 replace guralnik = 0 if (mmalone==3 | inlist(mmhs,2,3,4) | mmsaf==2 | mmwala==2 | inlist(mmtrya,2,3
)) & indager >=60
1612
1613 * Save dataset with a new name
1614 save datavariab03.dta
1615
1616 * WAVE 2, 4, 5, 6, 7, 8, 9
1617 * [b44] Self-reported general health (reversed)
1618 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1619 replace hehelp = . if hehelp<0
1620 * Reverse the self-rated health variable
1621 revrs hehelp
1622
1623 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1624 * [b1-b6] (+ 2 omitted) Depressive symptoms
1625 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1626 * [b1] Whether respondent has felt depressed much of the time during the past week
1627 replace psceda = . if psceda<0
1628 * (1 omitted) Whether respondent felt everything they did during the past week was an effort
1629 replace pscedb = . if pscedb<0
1630 * [b2] Whether respondent felt their sleep was restless during the past week
1631 replace pscedc = . if pscedc<0
1632 * [b3] Whether respondent was happy much of the time during the past week (reversed)

```



```

1633 replace pscedd = . if pscedd<0
1634 * [b4] Whether respondent felt lonely must of the time during the past week
1635 replace pscede = . if pscede<0
1636 * [b5] Whether respondent enjoyed life much of the time during the past week (reversed)
1637 replace pscedf = . if pscedf<0
1638 * [b6] Whether respondent felt sad much of the time during the past week
1639 replace pscedg = . if pscedg<0
1640 * (1 omitted) Whether respondent could not get going much of the time during the past week
1641 replace pscedh = . if pscedh<0
1642
1643 * Recode to the number 0 if participant answered "Yes"
1644 replace psceda = 0 if psceda == 1
1645 * Recode to the number 1 if participant answered "No"
1646 replace psceda = 1 if psceda == 2
1647
1648 * Recode to the number 0 if participant answered "Yes"
1649 replace pscedb = 0 if pscedb == 1
1650 * Recode to the number 1 if participant answered "No"
1651 replace pscedb = 1 if pscedb == 2
1652
1653 * Recode to the number 0 if participant answered "Yes"
1654 replace pscedc = 0 if pscedc == 1
1655 * Recode to the number 1 if participant answered "No"
1656 replace pscedc = 1 if pscedc == 2
1657
1658 * Recode to the number 0 if participant answered "Yes"
1659 replace pscede = 0 if pscede == 1
1660 * Recode to the number 1 if participant answered "No"
1661 replace pscede = 1 if pscede == 2
1662
1663 * Recode to the number 0 if participant answered "Yes"
1664 replace pscedg = 0 if pscedg == 1
1665 * Recode to the number 1 if participant answered "No"
1666 replace pscedg = 1 if pscedg == 2
1667
1668 * Recode to the number 0 if participant answered "Yes"
1669 replace pscedh = 0 if pscedh == 1
1670 * Recode to the number 1 if participant answered "No"
1671 replace pscedh = 1 if pscedh == 2
1672
1673 * Recode to the number 0 if participant answered "No"
1674 replace pscedd = 0 if pscedd == 2
1675 * Recode to the number 0 if participant answered "No"
1676 replace pscedf = 0 if pscedf == 2
1677
1678 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1679 * (3 omitted) Loneliness (reversed)
1680 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1681 * (1 omitted) How often respondent feels they lack companionship
1682 replace scfeela = . if scfeela<0
1683 * (1 omitted) How often respondent feels left out
1684 replace scfeelb = . if scfeelb<0
1685 * (1 omitted) How often respondent feels isolated from others
1686 replace scfeelc = . if scfeelc<0
1687
1688 * Reverse the negatively framed variables
1689 revrs scfeela
1690 revrs scfeelb
1691 revrs scfeelc
1692
1693 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1694 * Outcome variable - Quality-of-life
1695 * Replace variables as missing for any missing cases (coded as negative numbers in the ELSA dataset)

```

```

1696 replace scqola = . if scqola<0
1697 replace scqolb = . if scqolb<0
1698 replace scqolc = . if scqolc<0
1699 replace scqold = . if scqold<0
1700 replace scqole = . if scqole<0
1701 replace scqolf = . if scqolf<0
1702 replace scqolg = . if scqolg<0
1703 replace scqolh = . if scqolh<0
1704 replace scqoli = . if scqoli<0
1705 replace scqolj = . if scqolj<0
1706 replace scqolk = . if scqolk<0
1707 replace scqoll = . if scqoll<0
1708 replace scqolm = . if scqolm<0
1709 replace scqoln = . if scqoln<0
1710 replace scqolo = . if scqolo<0
1711 replace scqolp = . if scqolp<0
1712 replace scqolq = . if scqolq<0
1713 replace scqolr = . if scqolr<0
1714 replace scqols = . if scqols<0
1715
1716 * Recode each item into a 0-3 scale (13 of the 19 items were reversed)
1717 replace scqola = 0 if scqola == 1
1718 replace scqola = 1 if scqola == 2
1719 replace scqola = 2 if scqola == 3
1720 replace scqola = 3 if scqola == 4
1721 replace scqolb = 0 if scqolb == 1
1722 replace scqolb = 1 if scqolb == 2
1723 replace scqolb = 2 if scqolb == 3
1724 replace scqolb = 3 if scqolb == 4
1725 revrs scqolc
1726 replace revscqolc = 0 if revscqolc == 1
1727 replace revscqolc = 1 if revscqolc == 2
1728 replace revscqolc = 2 if revscqolc == 3
1729 replace revscqolc = 3 if revscqolc == 4
1730 replace scqold = 0 if scqold == 1
1731 replace scqold = 1 if scqold == 2
1732 replace scqold = 2 if scqold == 3
1733 replace scqold = 3 if scqold == 4
1734 revrs scqole
1735 replace revscqole = 0 if revscqole == 1
1736 replace revscqole = 1 if revscqole == 2
1737 replace revscqole = 2 if revscqole == 3
1738 replace revscqole = 3 if revscqole == 4
1739 replace scqolf = 0 if scqolf == 1
1740 replace scqolf = 1 if scqolf == 2
1741 replace scqolf = 2 if scqolf == 3
1742 replace scqolf = 3 if scqolf == 4
1743 revrs scqolg
1744 replace revscqolg = 0 if revscqolg == 1
1745 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 month
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
1820 b) five or six days a week
1821 replace scako = 2 if scako==1
1822 * Assign the number 1 if the participant reported having an alcoholic drink a) three or four days a
1823 week; or b) once or twice a week
1824 replace scako = 1 if inlist(scako,3,4)
1825
1826 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1827 * Socio-economic covariate - Education
1828 * Generate a new variable
1829 gen education = .
1830 * Assign the number 0 if the participant does not have any formal qualifications
1831 replace education = 0 if edqual==7
1832 * Assign the number 1 if the participant has A level equivalent, O level equivalent, or other grade
1833 equivalent
1834 replace education = 1 if inlist(edqual,3,4,5)
1835 * Assign the number 2 if the participant has completed some higher education (below degree), or has
1836 a degree or equivalent
1837 replace education = 2 if inlist(edqual,1,2)
1838
1839 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1840 * Demographic covariate - Ethnicity
1841 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1842 replace fqethnr = . if fqethnr<0
1843 * Assign the number 0 if the participant is White
1844 replace fqethnr = 0 if fqethnr==1
1845 * Assign the number 1 if the participant is Non-White
1846 replace fqethnr = 1 if fqethnr==2
1847
1848 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1849 * Demographic covariate - Biological sex
1850 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
1851 replace disex = . if disex<0
1852 * Assign the number 0 if the participant is male
1853 replace disex = 0 if disex==1
1854 * Assign the number 1 if the participant is female
1855 replace disex = 1 if disex==2
1856
1857 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1858 * Independent variable - Physical activity
1859 * Generate a new variable
1860 gen activity2 = .
1861 * Assign the number 3 if the participant partakes in vigorous activity more than once a week or once
1862 a week
1863 replace activity2 = 3 if heacta==1 | heacta==2
1864 * Assign the number 2 if the participant partakes in moderate activity more than once a week or once
1865 a week, and takes part in vigorous activity less than once a week
1866 replace activity2 = 2 if (heactb==1 | heactb==2) & inlist(heacta,3,4)
1867 * Assign the number 1 if the participant partakes in mild activity more than once a week or once a
1868 week, and takes part in moderate and vigorous activities less than once a week
1869 replace activity2 = 1 if (heactc==1 | heactc==2) & inlist(heacta,3,4) & inlist(heactb,3,4)
1870 * Assign the number 0 if the participant does not take part in activity of any intensity once a week
1871 or more
1872 replace activity2 = 0 if inlist(heacta,3,4) & inlist(heactb,3,4) & inlist(heactc,3,4)
1873 * Replace the variable as missing for participants with missing cases on all three variables
1874 replace activity2 = . if inlist(heacta,.) & inlist(heactb,.) & inlist(heactc,.)
1875
1876 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
1877 * Independent variable - Smoking status
1878 * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
1879 wave) variable at Wave 2
1880 gen heske2 = heske if wave==2
1881 * 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
1912 bysort idauniq: carryforward heske5, replace
1913 * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
values of the time variable "wave"
1914 gsort idauniq - wave
1915 * Carryforward (in a backward way) observations with respect to the time variable "wave" by
participant ID
1916 bysort idauniq: carryforward heske5, replace
1917
1918 * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
wave) variable at Wave 6
1919 gen heske6 = heske if wave==6
1920 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1921 tsset idauniq wave

```

```

1922 * Generate a completely balanced dataset
1923 tsfill, full
1924 * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 6 to the
follow-up waves) by participant ID
1925 bysort idauniq: carryforward heske6, replace
1926 * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
values of the time variable "wave"
1927 gsort idauniq - wave
1928 * Carryforward (in a backward way) observations with respect to the time variable "wave" by
participant ID
1929 bysort idauniq: carryforward heske6, replace
1930
1931 * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
wave) variable at Wave 7
1932 gen heske7 = heske if wave==7
1933 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1934 tsset idauniq wave
1935 * Generate a completely balanced dataset
1936 tsfill, full
1937 * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 7 to the
follow-up waves) by participant ID
1938 bysort idauniq: carryforward heske7, replace
1939 * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
values of the time variable "wave"
1940 gsort idauniq - wave
1941 * Carryforward (in a backward way) observations with respect to the time variable "wave" by
participant ID
1942 bysort idauniq: carryforward heske7, replace
1943
1944 * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
wave) variable at Wave 8
1945 gen heske8 = heske if wave==8
1946 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1947 tsset idauniq wave
1948 * Generate a completely balanced dataset
1949 tsfill, full
1950 * Carryforward observations with respect to the time variable "wave" (i.e., from Wave 8 to the
follow-up wave) by participant ID
1951 bysort idauniq: carryforward heske8, replace
1952 * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
values of the time variable "wave"
1953 gsort idauniq - wave
1954 * Carryforward (in a backward way) observations with respect to the time variable "wave" by
participant ID
1955 bysort idauniq: carryforward heske8, replace
1956
1957 * Generate a new variable duplicating the heske (reason disputed reported smoking from previous
wave) variable at Wave 9
1958 gen heske9 = heske if wave==9
1959 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
1960 tsset idauniq wave
1961 * Generate a completely balanced dataset
1962 tsfill, full
1963 * Sort data in memory by ascending values of the participant ID variable "idauniq" and descending
values of the time variable "wave"
1964 gsort idauniq - wave
1965 * Carryforward (in a backward way) observations with respect to the time variable "wave" by
participant ID
1966 bysort idauniq: carryforward heske9, replace
1967
1968 * Generate a new variable duplicating the bhesmkc (whether reported ever smoked cigarettes at Wave
1) variable at Wave 2
1969 gen bhesmkc = bhesmk

```



```

1970 * Assign the number 1 if the participant reported no longer smoking cigarettes by last interview
      (Wave 1)
1971 replace bhesmkc = 1 if heske==2 & wave==2
1972 * Assign the number 2 if the participant reported they
1973 replace bhesmkc = 2 if heske==1 & wave==2
1974
1975 * Generate a new variable and assign the number 0 if the participant reported never having smoked
      cigarettes
1976 gen smoking = 0 if hesmk==2
1977 * 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
1978 replace smoking = 0 if bhesmkc==2 & heska==2
1979 * Assign the number 1 if the participant reported having ever smoked cigarettes but reported that
      they do not smoke cigarettes at all nowadays
1980 replace smoking = 1 if (hesmk==1 | bhesmkc==1) & heska==2
1981 * Assign the number 2 if the participant reported smoking nowadays
1982 replace smoking = 2 if heska==1
1983
1984 * 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
1985 replace smoking = 1 if heske==3 & heska==2 & wave==2
1986 tab heskd
1987 * 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
1988 replace smoking = 0 if heske3==1 & heska==2 & wave==2
1989 * 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
1990 replace smoking = 1 if heske3==2 & heska==2 & wave==2
1991
1992 * 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
1993 replace smoking = 1 if heske==3 & heska==2 & wave==3
1994 * 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
1995 replace smoking = 0 if heske4==1 & heska==2 & wave==3
1996 * 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
1997 replace smoking = 1 if heske4==2 & heska==2 & wave==3
1998
1999 * 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
2000 replace smoking = 1 if heske==3 & heska==2 & wave==4
2001 * 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
2002 replace smoking = 0 if heske5==1 & heska==2 & wave==4
2003 * 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
2004 replace smoking = 1 if heske5==2 & heska==2 & wave==4
2005
2006 * 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
2007 replace smoking = 1 if heske==3 & heska==2 & wave==5
2008 * 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
2009 replace smoking = 0 if heske6==1 & heska==2 & wave==5
2010 * 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
2011 replace smoking = 1 if heske6==2 & heska==2 & wave==5
2012
2013 * 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
2014 replace smoking = 1 if heske==3 & heska==2 & wave==6
2015 * 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
2016 replace smoking = 0 if heske7==1 & heska==2 & wave==6
2017 * 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
2018 replace smoking = 1 if heske7==2 & heska==2 & wave==6
2019
2020 * 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
2021 replace smoking = 1 if heske==3 & heska==2 & wave==7
2022 * 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
2023 replace smoking = 0 if heske8==1 & heska==2 & wave==7
2024 * 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
2025 replace smoking = 1 if heske8==2 & heska==2 & wave==7
2026
2027 * 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
2028 replace smoking = 1 if heske==3 & heska==2 & wave==8
2029 * 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
2030 replace smoking = 0 if heske9==1 & heska==2 & wave==8
2031 * 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
2032 replace smoking = 1 if heske9==2 & heska==2 & wave==8
2033
2034 * 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
2035 replace smoking = 1 if heske==3 & heska==2 & wave==9
2036
2037 * Assign the number 1 if the participant reported that they stopped smoking cigarettes
2038 replace smoking = 1 if heskf==2
2039 * Assign the number 2 if the participant reported smoking cigarettes nowadays
2040 replace smoking = 2 if heskf==1
2041
2042 * Count total number of participants and observations
2043 unique idauniq
2044 * 15,022 individuals, 120,176 observations
2045
2046 * Save dataset with a new name
2047 save datavariables04.dta
2048
2049 * [b45-b48] Cognitive function
2050 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
2051 * [b45] Computed score from date questions (orientation in time)
2052 * Generate a new variable duplicating the cfdatd variable
2053 gen daymonth = cfdatd
2054 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2055 replace daymonth = . if daymonth<0
2056 * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2057 replace daymonth = 0 if daymonth==2
2058 * Generate a new variable duplicating the cfday variable
2059 gen day = cfday
2060 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2061 replace day = . if day<0
2062 * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2063 replace day = 0 if day==2
2064 * Generate a new variable duplicating the cfday variable
2065 gen year = cfday
2066 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2067 replace year = . if year<0
2068 * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2069 replace year = 0 if year==2

```

```

2070 * Generate a new variable duplicating the cfdatm variable
2071 gen month = cfdatm
2072 * Replace variable as missing for any missing cases (coded as negative numbers in the ELSA dataset)
2073 replace month = . if month<0
2074 * Assign the number 0 if the participant answered incorrectly or didn't know the answer
2075 replace month = 0 if month==2
2076 * Generate a new variable equal to the sum of the four orientation in time items to create a total
    score
2077 gen orientation = daymonth + day + year + month
2078 gen orientation2 = cfdscr if cfdscr>=0
2079 * Generate a new variable and assign the number 0 for participants who scored 0, 1, 2, or 3 points
    on the time orientation test
2080 gen oribi = 0 if inlist(orientation,0,1,2,3)
2081 * Assign the number 1 for participants who answered all questions correctly (i.e., scored 4) on the
    time orientation test
2082 replace oribi = 1 if orientation==4
2083
2084 * WAVE 2, 3, 4, 5, 6, 7, 8, 9
2085 * [b46-b47] Word-list learning (verbal learning and recall)
2086 tab cftest
2087 tab cfwrds
2088 * Generate a new variable duplicating the cflisen variable for participants with a score from 0 to 10
2089 gen learning = cflisen if cflisen>=0
2090 * Generate a new variable duplicating the cflisd variable for participants with a score from 0 to 10
2091 gen recall = cflisd if cflisd>=0
2092
2093 * [b46] Number of words recalled immediately
2094 sum learning
2095 * Assign the number 0 for participants with scores >1 standard deviation below the mean
2096 replace learning = 0 if learning >= 0 & learning < 3.990809
2097 * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2098 replace learning = 1 if learning >= 3.990809 & learning <= 7.640907
2099 * Assign the number 2 for participants with scores >1 standard deviation above the mean
2100 replace learning = 2 if learning > 7.640907 & learning != .
2101
2102 * [b47] Number of words recalled after delay
2103 sum recall
2104 * Assign the number 0 for participants with scores >1 standard deviation below the mean
2105 replace recall = 0 if recall >= 0 & recall < 2.353383
2106 * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2107 replace recall = 1 if recall >= 2.353383 & recall <= 6.666215
2108 * Assign the number 2 for participants with scores >1 standard deviation above the mean
2109 replace recall = 2 if recall > 6.666215 & recall != .
2110
2111 * WAVE 2, 3, 4, 5, 7, 8, 9
2112 * [b48] Number of animals mentioned (verbal fluency)
2113 tab cfani
2114 * Generate a new variable duplicating the cfani variable for participants with a score of 0 or more
2115 gen fluency = cfani if cfani>=0
2116 sum fluency
2117 * Assign the number 0 for participants with scores >1 standard deviation below the mean
2118 replace fluency = 0 if fluency >= 0 & fluency < 13.60175
2119 * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2120 replace fluency = 1 if fluency >= 13.60175 & fluency <= 27.79781
2121 * Assign the number 2 for participants with scores >1 standard deviation above the mean
2122 replace fluency = 2 if fluency > 27.79781 & fluency != .
2123
2124 * WAVE 2, 4, 6, 8, 9
2125 * [b49] Grip strength
2126 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
    dataset)
2127 replace mmgsd1 = . if mmgsd1 < 0
2128 replace mmgsd2 = . if mmgsd2 < 0

```

```

2129 replace mmgsd3 = . if mmgsd3 < 0
2130 replace mmgsn1 = . if mmgsn1 < 0
2131 replace mmgsn2 = . if mmgsn2 < 0
2132 replace mmgsn3 = . if mmgsn3 < 0
2133 replace mmgsd1 = . if mmgsd1==99
2134 replace mmgsd2 = . if mmgsd2==99
2135 replace mmgsd3 = . if mmgsd3==99
2136 replace mmgsn1 = . if mmgsn1==99
2137 replace mmgsn2 = . if mmgsn2==99
2138 replace mmgsn3 = . if mmgsn3==99
2139
2140 * Generate a new variable equal to the maximum grip strength across all available measures
2141 egen maxgrip = rowmax(mmgsd1 mmgsd2 mmgsd3 mmgsn1 mmgsn2 mmgsn3)
2142
2143 sum maxgrip
2144 * Assign the number 0 for participants with scores >1 standard deviation below the mean
2145 replace maxgrip = 0 if maxgrip >= 0 & maxgrip < 19.61438
2146 * Assign the number 1 for participants with scores ±1 standard deviation around the mean
2147 replace maxgrip = 1 if maxgrip >= 19.61438 & maxgrip <= 42.36316
2148 * Assign the number 2 for participants with scores >1 standard deviation above the mean
2149 replace maxgrip = 2 if maxgrip > 42.36316 & maxgrip != .
2150
2151 * Save dataset with a new name
2152 save datastop.dta
2153
2154 * [b50-b51] (+ 4 omitted) Biomarkers
2155 * WAVE 2, 4, 6, 8, 9
2156 * [b50] Blood fibrinogen level (g/L)
2157 tab cfib if cfib<0
2158 * Generate a new variable duplicating the cfib variable
2159 gen fibrinogen = cfib
2160 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2161 replace fibrinogen = . if fibrinogen < 0
2162 replace fibrinogen = . if fibrinogen > 9000
2163 sum fibrinogen
2164 * Assign the number 1 for participants with a fibrinogen level ≤4 g/L
2165 replace fibrinogen = 1 if fibrinogen <= 4
2166 * Assign the number 0 for participants with a fibrinogen level >4 g/L
2167 replace fibrinogen = 0 if fibrinogen > 4 & fibrinogen != .
2168
2169 * (1 omitted) Blood HDL level (mmol/L)
2170 tab hdl if hdl<0
2171 * Generate a new variable duplicating the hdl variable
2172 gen highdensity = hdl
2173 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2174 replace highdensity = . if highdensity < 0
2175 replace highdensity = . if highdensity > 9000
2176 sum highdensity
2177 * Assign the number 0 for participants with a HDL level <1 mmol/L
2178 replace highdensity = 0 if highdensity <1
2179 * Assign the number 1 for participants with a HDL level ≥1 mmol/L
2180 replace highdensity = 1 if highdensity >= 1 & highdensity != .
2181
2182 * (1 omitted) Blood triglyceride level (mmol/L)
2183 tab trig if trig<0
2184 * Generate a new variable duplicating the trig variable
2185 gen triglyceride = trig
2186 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2187 replace triglyceride = . if triglyceride < 0
2188 replace triglyceride = . if triglyceride > 9000

```

```

2189 sum triglyceride
2190 * Assign the number 1 for participants with a triglyceride level ≤2 mmol/L
2191 replace triglyceride = 1 if triglyceride ≤ 2
2192 * Assign the number 0 for participants with a triglyceride level >2 mmol/L
2193 replace triglyceride = 0 if triglyceride > 2 & triglyceride != .
2194
2195 * (1 omitted) Blood LDL level (mmol/L)
2196 tab ldl if ldl<0
2197 * Generate a new variable duplicating the ldl variable
2198 gen lowdensity = ldl
2199 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2200 replace lowdensity = . if lowdensity < 0
2201 replace lowdensity = . if lowdensity > 9000
2202 sum lowdensity
2203 * Assign the number 1 for participants with a LDL level ≤4 mmol/L
2204 replace lowdensity = 1 if lowdensity ≤ 4
2205 * Assign the number 0 for participants with a LDL level >4 mmol/L
2206 replace lowdensity = 0 if lowdensity > 4 & lowdensity != .
2207
2208 * [b51] Blood CRP level (mg/L)
2209 tab hscrp if hscrp<0
2210 * Generate a new variable duplicating the hscrp variable
2211 gen CRP = hscrp
2212 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2213 replace CRP = . if CRP < 0
2214 replace CRP = . if CRP > 9000
2215 sum CRP
2216 * Replace variable as missing for CRP values >20 mg/L
2217 replace CRP = . if CRP > 20
2218 * Assign the number 1 for participants with a CRP level ≤3 mg/L
2219 replace CRP = 1 if CRP ≤ 3
2220 * Assign the number 0 for participants with a CRP level >3 mg/L
2221 replace CRP = 0 if CRP > 3 & CRP != .
2222
2223 * (1 omitted) Blood glyated haemoglobin level (%)
2224 * Wave 2, 4 = %, Wave 6, 8, 9 = mmol/mol
2225 tab hba1c if hba1c<0
2226 * Generate a new variable duplicating the hba1c variable
2227 gen glyated = hba1c
2228 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)
2229 replace glyated = . if glyated < 0
2230 replace glyated = . if glyated > 9000
2231 sum glyated if wave==2
2232 sum glyated if wave==4
2233 sum glyated if wave==6
2234 sum glyated if wave==8
2235 sum glyated if wave==9
2236 * Transform HbA1c levels (mmol/mol) in Wave 6, 8, and 9 into % to match data in Wave 2 and 4
2237 replace glyated = ((glyated/10.929)+2.15) if inlist(wave,6,8,9)
2238 sum glyated if wave==2
2239 sum glyated if wave==4
2240 sum glyated if wave==6
2241 sum glyated if wave==8
2242 sum glyated if wave==9
2243 * Assign the number 1 for participants with a HbA1c level <6.5 %
2244 replace glyated = 1 if glyated < 6.5
2245 * Assign the number 0 for participants with a HbA1c level ≥6.5 %
2246 replace glyated = 0 if glyated ≥ 6.5 & glyated != .
2247
2248 * Save dataset with a new name

```

```

2249 save alldataefa.dta
2250
2251 *****
2252 ***DATA ANALYSIS***
2253 *****
2254
2255 * Keep variables required for analyses
2256 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 repctest gaittest revhehelf psceda pscedb pscedc
pscedd pscede pscedf pscedg pscedh revscfeela revscfeelb revscfeelc oribi learning recall fluency
maxgrip fibrinogen highdensity triglyceride lowdensity CRP glycated
2257 * Rename variables to more convenient forms
2258 rename revscacta scactarev
2259 rename revscactb scactbrev
2260 rename revscactc scactcrev
2261 rename revscactd scactdrev
2262 rename revhehelf hehelfrev
2263 rename revscfeela scfeelarev
2264 rename revscfeelb scfeelbrev
2265 rename revscfeelc scfeelcrev
2266 * Save dataset with a new name
2267 save efanew.dta
2268 * Keep data from Wave 2 only
2269 keep if wave==2
2270 * Drop unnecessary variable
2271 drop wave
2272 * Count total number of participants
2273 unique idauniq
2274 * 15,022 individuals
2275 * Save dataset with a new name
2276 save baseline.dta
2277 * Generate a new variable equal to the sum of variables with available "non-missing" data for each
participant
2278 egen nmcount = rownonmiss(_all), strok
2279 tab nmcount
2280 * Drop observation if > 25% missing values across the 64 variables of interest (idauniq and wave
data are complete for all observations)
2281 drop if nmcount<49
2282 * Drop unnecessary variable
2283 drop nmcount
2284 * Count total number of participants
2285 unique idauniq
2286 * 7,660 individuals
2287 * Generate a new variable equal to the sum of variables with missing data for each participant
2288 egen nmcount = rowmiss(_all)
2289 tab nmcount
2290 * Drop unnecessary variable
2291 drop nmcount
2292 * Set the seed
2293 set seed 1234
2294 * Generate random numbers
2295 gen random = uniform()
2296 sort random
2297 * Assign 30% of the total sample to the validation sub-sample
2298 gen byte validation = _n <= 2298
2299 * Save dataset with a new name
2300 save efatotal.dta
2301 * Keep participants assigned to the developmental sub-sample
2302 keep if validation == 0
2303 * Count total number of participants
2304 unique idauniq

```



```

2305 * 5,362 individuals
2306 * Save developmental dataset
2307 save developmentalnew.dta
2308 * Use efatotal.dta dataset
2309 use efatotal.dta
2310 * Keep participants assigned to the validation sub-sample
2311 keep if validation == 1
2312 * Count total number of participants
2313 unique idauniq
2314 * 2,298 individuals
2315 * Save validation dataset
2316 save validationnew.dta
2317
2318 * Use developmental dataset
2319 use developmentalnew.dta
2320 * Convert Stata data into a data file and Mplus input file
2321 stata2mplus using developmentalfinal.dta
2322
2323 * Use validation dataset
2324 use validationnew.dta
2325 * Convert Stata data into a data file and Mplus input file
2326 stata2mplus using validationfinal.dta
2327
2328 * Use efanew.dta dataset
2329 use efanew.dta
2330 * Drop variables omitted from final metric
2331 drop pscedb pscedh headlmo scfeelarev scfeelbrev scfeelcrev balance headlme triglyceride lowdensity
highdensity glycated scorg06
2332 * Save dataset with a new name
2333 save MLIRT.dta
2334 * Count total number of participants and observations
2335 unique idauniq
2336 * 15,022 individuals, 120,176 observations
2337 * Generate a new variable equal to the sum of variables with available "non-missing" data for each
participant
2338 egen nmcount = rownonmiss(_all), strok
2339 tab nmcount
2340 * Drop observation if > 50% missing values across the 51 variables of interest (idauniq and wave
data are complete for all observations)
2341 keep if nmcount>=27.5
2342 * Drop unnecessary variable
2343 drop nmcount
2344 * Count total number of participants and observations
2345 unique idauniq
2346 * 14,755 individuals, 66,133 observations
2347 * Generate a new variable equal to the sum of variables with missing data for each participant
2348 egen nmcount = rowmiss(_all)
2349 tab nmcount
2350 * Drop unnecessary variable
2351 drop nmcount
2352 * Save dataset with a new name
2353 save MLIRTtouse.dta
2354 * Export Stata data to .csv file
2355 export delimited using "", nolabel replace
2356
2357 * Use alldataefa.dta dataset
2358 use alldataefa.dta
2359
2360 * WAVE 2, 3, 4, 5, 6, 8, 9
2361 * Socio-economic covariate - Occupational class
2362 * Excluded Never worked and long-term unemployed
2363 * Replace variable as missing for any missing cases (coded as negative numbers or 99 in the ELSA
dataset)

```

```

2364 replace nssec8 = . if nssec8<0
2365 replace nssec8 = . if nssec8 == 99
2366 * Generate a new variable
2367 gen mynssec3 = .
2368 * 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
2369 replace mynssec3 = 2 if inlist(nssec8,1,2)
2370 * 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
2371 replace mynssec3 = 1 if inlist(nssec8,3,4)
2372 * 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
2373 replace mynssec3 = 0 if inlist(nssec8,5,6,7)
2374 * Overwrite dataset, by replacing the previously saved file
2375 save alldataefa.dta, replace
2376
2377 * Keep variables required for analyses
2378 keep idauniq wave indager fqethnr education totwq5_bu_s disex scako smoking activity2 mynssec3 QoL
2379 * Save dataset with a new name
2380 save MLIRTGMM.dta
2381 * One-to-one merge of data in memory with MLIRTdata.dta (exported from RStudio) on participant ID
2382 merge 1:1 idauniq wave using MLIRTdata.dta
2383 * Sort from lowest to highest participant ID
2384 sort idauniq
2385 * Save dataset with a new name
2386 save MLIRTdatafull.dta
2387 tab AHA
2388 * Drop observations with missing AHA scores
2389 drop if AHA==.
2390 * Count total number of participants and observations
2391 unique idauniq
2392 * 14,755 individuals, 66,133 observations
2393 * Save dataset with a new name
2394 save MLIRTfullAHA.dta
2395 sum indager, d
2396 sum indager if wave==2, d
2397 * 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
2398 bysort idauniq (wave): gen obsnr = _n
2399 * Generate a variable that assigns the number of total observations to each row of data for a given
participant
2400 bysort idauniq: gen obscount = _N
2401 tab obscount
2402 * Include participants with AHA scores in at least two waves
2403 keep if obscount!=1
2404 * Count total number of participants and observations
2405 unique idauniq
2406 * 11,739 individuals, 63,117 observations
2407 * Keep necessary variables
2408 keep idauniq wave indager fqethnr education totwq5_bu_s disex scako smoking activity2 mynssec3 QoL
AHA obsnr obscount
2409 sum idauniq if obsnr==1 & wave==2
2410 * Generate a variable that assigns the number 1 to the row representing participants' first
observation if this corresponds to Wave 2 (baseline)
2411 bysort idauniq (wave): gen firstwave = 1 if obsnr==1 & wave==2
2412 * Carry the value of this last variable forwards to the remainder of a participant's observations
2413 bysort idauniq: gen firstwave_cons = firstwave[1]
2414 * Include participants with AHA scores in at least two waves, including at baseline
2415 keep if firstwave_cons==1
2416 * Count total number of participants and observations
2417 unique idauniq
2418 * 7,416 individuals, 42,716 observations

```

```

2419
2420 * Generate a new variable duplicating the biological sex variable at Wave 2
2421 gen sex = disex if wave==2
2422 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2423 tsset idauniq wave
2424 * 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
2425 bysort idauniq: carryforward sex, replace
2426
2427 * Generate a new variable duplicating the education variable at Wave 2
2428 gen qualifications = education if wave==2
2429 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2430 tsset idauniq wave
2431 * 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
2432 bysort idauniq: carryforward qualifications, replace
2433
2434 * Generate a new variable duplicating the wealth variable at Wave 2
2435 gen wealth = totwq5_bu_s if wave==2
2436 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2437 tsset idauniq wave
2438 * 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
2439 bysort idauniq: carryforward wealth, replace
2440
2441 * Generate a new variable duplicating the age variable at Wave 2
2442 gen age = indager if wave==2
2443 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2444 tsset idauniq wave
2445 * 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
2446 bysort idauniq: carryforward age, replace
2447
2448 * Generate a new variable duplicating the ethnicity variable at Wave 2
2449 gen ethnicity = fqethnr if wave==2
2450 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2451 tsset idauniq wave
2452 * 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
2453 bysort idauniq: carryforward ethnicity, replace
2454
2455 * Generate a new variable duplicating the occupational class variable at Wave 2
2456 gen nssec3 = mynssec3 if wave==2
2457 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2458 tsset idauniq wave
2459 * 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
2460 bysort idauniq: carryforward nssec3, replace
2461
2462 * Save dataset with a new name
2463 save GMMcovdataSES.dta
2464 * Keep variables required for analyses
2465 keep idauniq wave sex qualifications wealth age ethnicity nssec3 QoL AHA
2466 * Reshape data into wide format for observations identified by participant ID and add wave as an
identifying time period
2467 reshape wide QoL AHA, j(wave) i(idauniq)
2468 * Save dataset with a new name
2469 save GMMcovdatawideSES.dta
2470 sum QoL9, d
2471 * Generate a new variable and assign the number 0 if a participant's quality-of-life score is below
the sample median
2472 gen QoL9binary = 0 if QoL9<43
2473 * Assign the number 1 if a participant's quality-of-life score is above or equal to the sample median

```

```

2474 replace QoL9binary = 1 if QoL9 >=43 & QoL9!=.
2475 * Save dataset with a new name
2476 save GMMcovbinarySES.dta
2477
2478 * Dummy variables for conditional GMM
2479 * Education
2480 * 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)
2481 gen medumed = 0 if inlist(qualifications,0,2)
2482 replace medumed = 1 if qualifications == 1
2483 * High education (coded as 1) versus low or medium education (coded as 0)
2484 gen highed = 0 if inlist(qualifications,0,1)
2485 replace highed = 1 if qualifications == 2
2486 * Occupational class
2487 * Intermediate occupations (coded as 1) versus lower or higher occupations (coded as 0)
2488 gen mediumocc = 0 if inlist(nssec3,0,2)
2489 replace mediumocc = 1 if nssec3 == 1
2490 * Higher occupations (coded as 1) versus lower or intermediate occupations (coded as 0)
2491 gen highocc = 0 if inlist(nssec3,0,1)
2492 replace highocc = 1 if nssec3 == 2
2493 * Wealth
2494 * 2nd quintile (coded as 1) versus 1st, 3rd, 4th, or 5th quintile (coded as 0)
2495 gen quint2 = 0 if inlist(wealth,1,3,4,5)
2496 replace quint2 = 1 if wealth == 2
2497 * 3rd quintile (coded as 1) versus 1st, 2nd, 4th, or 5th quintile (coded as 0)
2498 gen quint3 = 0 if inlist(wealth,1,2,4,5)
2499 replace quint3 = 1 if wealth == 3
2500 * 4th quintile (coded as 1) versus 1st, 2nd, 3rd, or 5th quintile (coded as 0)
2501 gen quint4 = 0 if inlist(wealth,1,2,3,5)
2502 replace quint4 = 1 if wealth == 4
2503 * 5th quintile (coded as 1) versus 1st, 2nd, 3rd, or 4th quintile (coded as 0)
2504 gen quint5 = 0 if inlist(wealth,1,2,3,4)
2505 replace quint5 = 1 if wealth == 5
2506 * Save dataset with a new name
2507 save GMMcovbinarySESdu.dta
2508 * Convert Stata data into a data file and Mplus input file
2509 stata2mplus using GMMcovbinarySESdu.dta
2510
2511 * Use MLIRTfullAHA.dta dataset
2512 use MLIRTfullAHA.dta
2513 * Count total number of participants and observations
2514 unique idauniq
2515 * 14,755 individuals, 66,133 observations
2516 * 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
2517 bysort idauniq (wave): gen obsnr = _n
2518 * Generate a new variable duplicating the biological sex variable at Wave 2
2519 gen sex = disex if obsnr==1
2520 * Declare a panel dataset with participant ID "idauniq" and time variable "wave"
2521 tsset idauniq wave
2522 * 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
2523 bysort idauniq: carryforward sex, replace
2524 * Keep variables required for Receiver Operating Characteristic (ROC) analyses
2525 keep idauniq wave sex QoL AHA
2526 * Save dataset with a new name
2527 save ROCdata.dta
2528 sum QoL, d
2529 * Generate a new variable and assign the number 0 if a participant's quality-of-life score is below
the sample median
2530 gen QoLbinary = 0 if QoL < 43
2531 * Assign the number 1 if a participant's quality-of-life score is above or equal to the sample median
2532 replace QoLbinary = 1 if QoL >= 43 & QoL !=.

```

```

2533 * Overwrite dataset, by replacing the previously saved file
2534 save ROCdata.dta, replace
2535 * Drop unnecessary variable
2536 drop QoL
2537 * Reshape data into wide format for observations identified by participant ID and add wave as an
identifying time period
2538 reshape wide AHA QoLbinary, j(wave) i (idauniq)
2539 * Save dataset with a new name
2540 save ROC92wide.dta
2541
2542 * ROC analyses, clustering at the participant level and adjusting the control distribution for
biological sex
2543 * Quality-of-life at Wave 3
2544 rocreg QoLbinary3 AHA2, probit ml ctrlcov(sex) ctrlmodel(linear) cluster(idauniq)
2545 * Quality-of-life at Wave 6
2546 rocreg QoLbinary6 AHA2, probit ml ctrlcov(sex) ctrlmodel(linear) cluster(idauniq)
2547 * Quality-of-life at Wave 9
2548 rocreg QoLbinary9 AHA2, probit ml ctrlcov(sex) ctrlmodel(linear) cluster(idauniq)
2549
2550 * Use MLIRTfullAHA.dta dataset
2551 use MLIRTfullAHA.dta
2552 * Two-level mixed-effects linear regression of AHA scores on lifestyle behaviours (adjusted for
covariates), with random intercepts by participant ID
2553 mixed AHA indager i.disex i.fqethnr i.scako i.smoking i.activity2 || idauniq:
2554 * Store estimates for later use
2555 estimates store randint
2556 * Fit indices
2557 estimates stats
2558 * 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
2559 mixed AHA indager i.disex i.fqethnr i.scako i.smoking i.activity2 || idauniq: wave
2560 * Store estimates for later use
2561 estimates store randslope
2562 * Fit indices
2563 estimates stats
2564 * Likelihood ratio test
2565 lrtest randslope randint
2566
2567 * Use MLIRTfullAHA.dta dataset
2568 use MLIRTfullAHA.dta
2569 * Count total number of participants and observations
2570 unique idauniq
2571 * 14,755 individuals, 66,133 observations
2572 * 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
2573 bysort idauniq (wave): gen obsnr = _n
2574 * Generate a variable that assigns the number 1 to the row representing participants' first
observation if this corresponds to Wave 2 (baseline)
2575 bysort idauniq (wave): gen firstwave = 1 if obsnr==1 & wave==2
2576 * Carry the value of this last variable forwards to the remainder of a participant's observations
2577 bysort idauniq: gen firstwave_cons = firstwave[1]
2578 * Include participants with at least some data at baseline
2579 keep if firstwave_cons==1
2580 * Generate a variable that assigns the number of total observations to each row of data for a given
participant
2581 bysort idauniq: gen obscount = _N
2582 tab pycount
2583 * Count total number of participants and observations
2584 unique idauniq
2585 * 8,670 individuals, 43,970 observations
2586 * Keep necessary variables
2587 keep idauniq wave indager fqethnr education totwq5_bu_s disex scako smoking activity2 mynssec3 QoL
AHA obsnr obscount firstwave firstwave_cons

```



```

2588 * Generate a new variable and assign the number 0 to participants with data at a single wave
2589 gen total = 0 if obscount == 1
2590 * Assign the number 1 to participants with AHA scores in at least two waves, including baseline
2591 replace total = 1 if obscount >1
2592 * Keep data from Wave 2 only
2593 keep if wave==2
2594 * Count total number of participants
2595 unique idauniq
2596 * 8,670 individuals
2597 tab total
2598 * Baseline characteristics of the complete-case sample used for the GMM analyses versus the sample
    excluded due to missing data (unweighted)
2599 sum indager if total == 0
2600 sum indager if total == 1
2601 tab fqethnr if total == 0
2602 tab fqethnr if total == 1
2603 tab education if total == 0
2604 tab education if total == 1
2605 tab totwq5_bu_s if total == 0
2606 tab totwq5_bu_s if total == 1
2607 tab disex if total == 0
2608 tab disex if total == 1
2609 tab scako if total == 0
2610 tab scako if total == 1
2611 tab smoking if total == 0
2612 tab smoking if total == 1
2613 tab activity2 if total == 0
2614 tab activity2 if total == 1
2615 tab mynssec3 if total == 0
2616 tab mynssec3 if total == 1
2617 ttest indager, by(total)
2618 tabulate fqethnr total, chi2
2619 tabulate education total, chi2
2620 tabulate totwq5_bu_s total, chi2
2621 tabulate disex total, chi2
2622 tabulate scako total, chi2
2623 tabulate smoking total, chi2
2624 tabulate activity2 total, chi2
2625 tabulate mynssec3 total, chi2
2626
2627 * Use GMMcovdataSES.dta dataset
2628 use GMMcovdataSES.dta
2629 * Drop observations with missing data on socio-economic or demographic covariates
2630 drop if sex==. | qualifications==. | wealth==. | age==. | ethnicity==. | nssec3 ==.
2631 * Count total number of participants and observations
2632 unique idauniq
2633 * 6,539 individuals, 37,768 observations
2634 * Drop unnecessary variables
2635 drop obsnr obscount
2636 * 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
2637 bysort idauniq (wave): gen obsnr = _n
2638 * Generate a variable that assigns the number of total observations to each row of data for a given
    participant
2639 bysort idauniq: gen obscount = _N
2640 tab wave
2641 * Summarise number of observations per participant
2642 tab obscount if wave==2
2643 sum obscount if wave==2
2644 * Keep necessary variables
2645 keep idauniq wave sex qualifications wealth age ethnicity nssec3 AHA QoL scako smoking activity2
2646 * Reshape data into wide format for observations identified by participant ID and add wave as an
    identifying time period

```

```

2647 reshape wide QoL AHA scako smoking activity2, j(wave) i(idauniq)
2648 * Save dataset with a new name
2649 save descbyclassGMMcovdataSESwide.dta
2650
2651 * Import posterior probabilities of class membership following the conditional 3-class GMM
2652 import excel "", sheet("") firstrow
2653 * Save dataset with a new name
2654 save classesfinal.dta
2655 * Use descbyclassGMMcovdataSESwide.dta dataset
2656 use descbyclassGMMcovdataSESwide.dta
2657 * One-to-one merge of data in memory with classesfinal.dta on participant ID
2658 merge 1:1 idauniq using classesfinal.dta
2659 * Save dataset with a new name
2660 save NEWclassdescfinal.dta
2661
2662 * Summarise the AHA scores at each wave
2663 sum AHA2, d
2664 sum AHA3, d
2665 sum AHA4, d
2666 sum AHA5, d
2667 sum AHA6, d
2668 sum AHA7, d
2669 sum AHA8, d
2670 sum AHA9, d
2671
2672 * Summarise posterior probabilities for each latent class
2673 sum posterior1 if class==1
2674 sum posterior2 if class==2
2675 sum posterior3 if class==3
2676
2677 * Descriptive statistics stratified by class membership
2678 sum age if class==1
2679 sum age if class==2
2680 sum age if class==3
2681
2682 tab sex if class==1
2683 tab sex if class==2
2684 tab sex if class==3
2685
2686 tab ethnicity if class==1
2687 tab ethnicity if class==2
2688 tab ethnicity if class==3
2689
2690 tab wealth if class==1
2691 tab wealth if class==2
2692 tab wealth if class==3
2693
2694 tab qualifications if class==1
2695 tab qualifications if class==2
2696 tab qualifications if class==3
2697
2698 tab nssec3 if class==1
2699 tab nssec3 if class==2
2700 tab nssec3 if class==3

```