

```
70 * program SCF01xj.SAS separates the 'X' variables from the 'J' variables;  
71 * revised version keeps only x's used in later programs;  
72 * Karl's program (based on Karen Pence's program), 5/16/03;  
73  
74 libname mydisk 'd:\scfs\2001';  
NOTE: Libref MYDISK was successfully assigned as follows:  
Engine: V7  
Physical Name: d:\scfs\2001  
75  
76 options obs=max;  
77  
78 proc cimport infile='d:\scfs\2001\scf2001' data='d:\scfs\2001\test2001';  
WARNING: The Base Product product with which CIMPORT is associated will expire within 30 days.  
Please contact your SAS installation representative to have it renewed.  
79
```

NOTE: Proc CIMPORT begins to create/update data set MYDISK.TEST2001
NOTE: Data set contains 5309 variables and 22210 observations.
Logical record length is 27816

NOTE: PROCEDURE CIMPORT used:
real time 8:29.97
cpu time 1:36.63

```
80 data mydisk.small2001;  
WARNING: The Base Product product with which DATASTEP is associated will expire within 30 days.  
Please contact your SAS installation representative to have it renewed.
```

```
81 set mydisk.test2001;  
82 keep  
83  
84 Y1 YY1 X42000 X42001 X14 X19  
85 X101-X105 X108 X110 X114 X116 X120 X122 X126 X128 X132 X134 X202 X204  
86 X208 X210 X214 X216 X220 X222 X301 X302 X305 X308 X312 X316 X320 X324  
87 X328 X407 X408 X409 X410 X411  
88 X413 X421 X424 X427 X430 X507 X513 X526 X604 X607 X614 x617 X623 X635  
89 X701 X716 X717 X805 X905  
90  
91 X1005 X1044 X1103 X1108 X1114 X1119 X1125 X1130 X1136 X1215 X1219 X1405  
92 X1409 X1417 X1505 X1509 X1517 X1605 X1617 X1619 X1621 X1703 X1705  
93 X1706 X1709 X1715 X1729 X1803 X1805 X1806 X1809 X1815 X1829 X1903 X1905  
94 X1906 X1909 X1915 X1929  
95  
96 X2002 X2003 X2006 X2009 X2012 X2013 X2016 X2019 X2202 X2218 X2318 X2418  
97 X2422 X2424 X2506 X2519 X2606 X2619 X2623 X2625 X2710 X2723 X2727 X2740  
98 X2810 X2823 X2827 X2840 X2910 X2923 X2927 X2940  
99  
100 X3006-X3020 X3023 X3119 X3121  
101 X3122 X3124 X3126 X3127 X3129 X3130 X3219 X3221 X3222 X3224 X3226 X3227  
102 X3229 X3230 X3319 X3321 X3322 X3324 X3326 X3327 X3329 X3330 X3335 X3336  
103 X3407-X3409 X3411-X3413 X3415-X3417 X3419-X3421 X3423-X3425 X3427-X3429  
104 X3506-X3507 X3510-X3511 X3514-X3515 X3518-X3519 X3522-X3523 X3526-X3527  
105 X3529 X3601 X3606-X3610 X3616-X3620 X3626-X3630 X3706 X3711  
106 X3716 X3718 X3721 X3804 X3807 X3810 X3813 X3816 X3818 X3819-X3835 X3902  
107 X3906 X3908 X3910 X3912 X3913-X3928 X3930 X3932 X3942  
108  
109 X4006 X4010 X4018  
110 X4020 X4022 X4024 X4026 X4028 X4030 X4032 X4100 X4106 X4110 X4112 X4113  
111 X4117 X4131 X4132 X4135 X4137 X4140 X4202-X4204 X4216 X4217 X4220 X4222  
112 X4224 X4226 X4227 X4229-X4231 X4302-X4304 X4316 X4317 X4320 X4322 X4324  
113 X4326 X4327 X4329-X4331 X4402-X4404 X4416 X4417 X4420 X4422 X4424 X4426  
114 X4427 X4429-X4431 X4436 X4700 X4511 X4512 X4706 X4710 X4712 X4713 X4717  
115 X4731 X4732 X4735 X4737 X4740 X4802-X4804 X4816 X4817 X4820 X4822 X4824  
116 X4826 X4827 X4829-X4831 X4902-X4904 X4916 X4917 X4920 X4922 X4924 X4926  
117 X4927 X4929-X4931  
118  
119 X5002-X5004 X5016 X5017 X5020 X5022 X5024 X5026 X5027  
120 X5029-X5031 X5036 X5111 X5112 X5306 X5307 X5311 X5312 X5318 X5326 X5334  
121 X5418 X5426 X5434 X5319 X5327 X5335 X5419 X5427 X5435 X5604 X5612 X5620  
122 X5628 X5636 X5644 X5608 X5616 X5624 X5632 X5640 X5648 X5609 X5617 X5625  
123 X5633 X5641 X5649 X5702  
124 X5704 X5706 X5708 X5710 X5712 X5714 X5716 X5718 X5720 X5722 X5724-X5727  
125 X5729 X5733-X5743 X5801-X5821 X5823 X5824 X5825 X5901 X5902 X5904 X5905  
126 X5909 X5910 X5915 X6030 X6101 X6105  
127 X6124 X6329  
128 X6772 X6773 X6774 X6809 X6810 X6820 X6835  
129  
130 X7018 X7133-X7134 X7169 X7179 X7183 X7372 X7381 X7396 X7402 X7412 X7583  
131 X7584 X7585 X7575 X7633-X7634 X7662 X7824 X7847 X7870 X7924 X7947 X7970
```

```

132
133 X8007 X8020-X8022 X8023 X8166 X8167 X8168 X8188
134
135 X9113-X9118 X9131-X9133
136 ;
137
138 run;

```

WARNING: Not all variables in the list X3006-X3020 were found.
WARNING: The variable X3912 in the DROP, KEEP, or RENAME list has never been referenced.
WARNING: The variable X3942 in the DROP, KEEP, or RENAME list has never been referenced.
WARNING: Not all variables in the list X5724-X5727 were found.
WARNING: Not all variables in the list X5801-X5821 were found.
WARNING: The variable X5909 in the DROP, KEEP, or RENAME list has never been referenced.
WARNING: The variable X5915 in the DROP, KEEP, or RENAME list has never been referenced.
NOTE: The data set MYDISK.SMALL2001 has 22210 observations and 526 variables.
NOTE: DATA statement used:
real time 4:45.39
cpu time 10.70 seconds

```

139 * networth01.sas, calculates net worth for the 2001 SCF;
140 * This program updates FRBNW95.SAS, which is the Federal-Reserve-provided
141 definitions of net worth from the 1995 SCF codebook.;
142 * 5/16/03;
143

```

```

144 libname mydisk 'd:\scfs\2001';
NOTE: Libref MYDISK was successfully assigned as follows:
Engine: V7
Physical Name: d:\scfs\2001
145 options obs=max;
146

```

147 data mydisk.small2001nw;
WARNING: The Base Product product with which DATASTEP is associated will expire within 30 days.
Please contact your SAS installation representative to have it renewed.

```

148 set mydisk.small2001;
149
150 scfwt=(x42000/5);
151 scfwt2 = (x42001/5);
152
153 * -----;
154 * -----;
155 * SAS CODE TO DEFINE NET WORTH: ONE POSSIBLE DEFINITION;
156 * -----;
157 * -----;
158 ***** Financial assets *****;
159
160 * checking accounts other than money market;
161 CHECKING =MAX(0, X3506)*(X3507=5)+MAX(0, X3510)*(X3511=5)+
162 MAX(0, X3514)*(X3515=5)+MAX(0, X3518)*(X3519=5)+
163 MAX(0, X3522)*(X3523=5)+MAX(0, X3526)*(X3527=5)+
164 MAX(0, X3529)*(X3527=5);
165
166 * savings accounts;
167 SAVI NG = MAX(0, X3804)+MAX(0, X3807)+MAX(0, X3810)+MAX(0, X3813)
168 +MAX(0, X3816)+MAX(0, X3818);
169
170 * money market deposit accounts;
171 MMDA= MAX(0, X3506)*((X3507=1)*(11<=X9113<=13))
172 +MAX(0, X3510)*((X3511=1)*(11<=X9114<=13))
173 +MAX(0, X3514)*((X3515=1)*(11<=X9115<=13))
174 +MAX(0, X3518)*((X3519=1)*(11<=X9116<=13))
175 +MAX(0, X3522)*((X3523=1)*(11<=X9117<=13))
176 +MAX(0, X3526)*((X3527=1)*(11<=X9118<=13))
177 +MAX(0, X3529)*((X3527=1)*(11<=X9118<=13))
178 +MAX(0, X3706)*(11<=X9131<=13)
179 +MAX(0, X3711)*(11<=X9132<=13)
180 +MAX(0, X3716)*(11<=X9133<=13)
181 +MAX(0, X3718)*(11<=X9133<=13);
182
183 * money market mutual funds;
184 MMMF= MAX(0, X3506)*(X3507=1)*(X9113<11|X9113>13)
185 +MAX(0, X3510)*(X3511=1)*(X9114<11|X9114>13)
186 +MAX(0, X3514)*(X3515=1)*(X9115<11|X9115>13)
187 +MAX(0, X3518)*(X3519=1)*(X9116<11|X9116>13)
188 +MAX(0, X3522)*(X3523=1)*(X9117<11|X9117>13)
189 +MAX(0, X3526)*(X3527=1)*(X9118<11|X9118>13)
190 +MAX(0, X3529)*(X3527=1)*(X9118<11|X9118>13)
191 +MAX(0, X3706)*(X9131<11|X9131>13)

```

```

192 +MAX(0, X3711)*(X9132<11|X9132>13)
193 +MAX(0, X3716)*(X9133<11|X9133>13)
194 +MAX(0, X3718)*(X9133<11|X9133>13);
195
196 * total money market accounts;
197 MMA=MMDA+MMMF;
198
199 * call accounts at brokerages;
200 CALL=MAX(0, X3930);
201
202 * transaction accounts;
203 LIQ=CHECKING+SAVING+MMA+CALL;
204
205 * certificates of deposit;
206 CDS = MAX(0, X3721);
207
208 * mutual funds;
209 * stock mutual funds;
210 STMUTF=(X3821=1)*MAX(0, X3822);
211 * tax-free bond mutual funds;
212 TFBMUTF=(X3823=1)*MAX(0, X3824);
213 * government bond mutual funds;
214 GBMUTF=(X3825=1)*MAX(0, X3826);
215 * other bond mutual funds;
216 OBMUTF=(X3827=1)*MAX(0, X3828);
217 * combination and other mutual funds;
218 COMUTF=(X3829=1)*MAX(0, X3830);
219
220 * total directly-held mutual funds, excluding MMMFs;
221 MMMF=STMUTF+TFBMUTF+GBMUTF+OBMUTF+COMUTF;
222
223 * stocks;
224 STOCKS = MAX(0, X3915);
225
226 * bonds, not including bond funds or savings bonds;
227 NOTXBND = X3910;
228 MORTBND = X3906;
229 GOVTBND = X3908;
230 OBND = X7634+X7633;
231
232 BOND = NOTXBND + MORTBND + GOVTBND + OBND;
233
234
235 * quasi-liquid retirement accounts (IRAs and thrift-type accounts);
236 * individual retirement accounts;
237 IRAKH = MAX(0, X3610)+MAX(0, X3620)+MAX(0, X3630);
238
239 * thrift-type plans (included if type is 401k, 403b, thrift,
240 savings, SRA, or if participant has option to borrow or
241 withdraw);
242 * PENEQ counts thrift amounts invested in stock;
243
244 ARRAY PTYPE{*} X4216 X4316 X4416 X4816 X4916 X5016;
245 ARRAY PAMT{*} X4226 X4326 X4426 X4826 X4926 X5026;
246 ARRAY PBOR{*} X4227 X4327 X4427 X4827 X4927 X5027;
247 ARRAY PWIT{*} X4231 X4331 X4431 X4831 X4931 X5031;
248 ARRAY PALL{*} X4234 X4334 X4434 X4834 X4934 X5034;
249
250 THRI FT = 0; PENEQ =0 ;
251 DO I=1 TO DIM(PTYPE);
252 THRI FT=THRI FT+PAMT{I}*(PTYPE{I}=1|PTYPE{I}=2|PBOR{I}=1|PWIT{I}=1);
253 PENEQ =PENEQ + PAMT{I}*(PTYPE{I}=1|PTYPE{I}=2|PBOR{I}=1|PWIT{I}=1)*
254 ((PALL{I}=1) + .5*(PALL{I}=3));
255 END;
256
257 * future pensions (accumulated in an account for the R/S);
258
259 FUTPEN=MAX(0, X5604)+MAX(0, X5612)+MAX(0, X5620)+MAX(0, X5628)+
260 MAX(0, X5636)+MAX(0, X5644);
261
262 * total quasi-liquid => sum of IRAs, thrift accounts, and future pensions;
263
264 * sum of IRAs and thrift accounts;
265 RETQLIQ=IRAKH+THRI FT+FUTPEN;
266
267 * savings bonds;
268 SAVBND = X3902;
269
270 * cash value of whole life insurance;

```

```

271 CASHLI=MAX(0,X4006);
272
273 * other managed assets;
274 * annuities w/equity interest;
275 ANNUIT=X6820;
276 * trusts & managed investment accounts w/equity interest;
277 TRUSTS=X6835;
278 OTHMA=ANNUIT+TRUSTS;
279
280 * other financial assets;
281 * includes loans, future proceeds, royalties, futures, non-public stock,
282 deferred compensation, oil/gas/mineral invest., cash n.e.c.;
283 OTHFIN = X4018+X4022*(X4020 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77)) +
284 X4026*(X4024 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77)) +
285 X4030*(X4028 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77));
286
287 FIN=LIQ+CDS+NMMF+STOCKS+BOND+RETQLIQ+SAVBND+CASHLI+OTHMA+OTHFIN;
288
289 ***** Nonfinancial assets *****;
290
291 * vehicles (includes autos, motorhomes, RVs, airplanes, boats);
292 VEHC=MAX(0,X8166)+MAX(0,X8167)+MAX(0,X8168)+MAX(0,X8188)+
293 MAX(0,X2422)+MAX(0,X2506)+MAX(0,X2606)+MAX(0,X2623);
294
295 * primary residence;
296 * X507 is maxed at 90%, as some cases report having 100%
297 of their property used for farming;
298
299 IF (X507 > 9000) THEN X507=9000;
300 HOUSES=SUM(0,X604,X614,X623,X716) + ((10000-X507)/10000)*(X513+X526);
301
302 IF (X701=-7 & X7133=1) THEN DO;
303 HOUSES=(X7134/10000)*X716;
304 END;
305
306
307
308 * other residential real estate;
309 * land contracts, Section E properties coded as 1-4 family residence,
310 misc vacation homes;
311 ORESRE = MAX(X1405,X1409)+MAX(X1505,X1509)+MAX(X1605,X1609) +
312 MAX(0,X1619) + (X1703 IN (12, 14, 21, 22, 25, 40, 41, 42, 43, 44, 49, 50,
313 52, 999))*MAX(0,X1706)*(X1705/10000) + (X1803 IN (12, 14, 21, 22,
314 25, 40, 41, 42, 43, 44, 49, 50, 52, 999))*MAX(0,X1806)*(X1805/10000) +
315 (X1903 IN (12, 14, 21, 22, 25, 40, 41, 42, 43, 44, 49, 50, 52, 999))*
316 MAX(0,X1906)*(X1905/10000) + MAX(0,X2002);
317
318 * net equity in nonresidential real estate;
319 * Section E, not coded as 1-4 family residence, misc property;
320 * nonresidential debt => includes Section E nonresidential, misc property,
321 and Section I apparently nonresidential (78);
322
323 NNRESRE =(X1703 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*MAX(0,X1706)*
324 (X1705/10000) + (X1803 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*
325 MAX(0,X1806)*(X1805/10000) +
326 (X1903 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*MAX(0,X1906)*(X1905/10000) +
327 MAX(0,X2012)
328 - (X1703 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*X1715*(X1705/10000)
329 - (X1803 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*X1815*(X1805/10000)
330 - (X1903 IN (10, 11, 13, 24, 45, 46, 47, 48, 51, -7))*X1915*(X1905/10000)
331 - X2016
332 - X2723*(X2710 IN (78)) - X2740*(X2727 IN (78))
333 - X2823*(X2810 IN (78)) - X2840*(X2827 IN (78))
334 - X2923*(X2910 IN (78)) - X2940*(X2927 IN (78));
335
336 * business interests;
337 * for active businesses, value is net equity if business were sold
338 today, plus loans from HH to business, minus loans from
339 business to HH not previously reported, plus value of personal
340 assets used as collateral for business loans that were reported earlier;
341 BUS = MAX(0,X3129) + MAX(0,X3124) - MAX(0,X3126)*(X3127=5) +
342 MAX(0,X3121)*(X3122=1|X3122=6) +
343 MAX(0,X3229) + MAX(0,X3224) - MAX(0,X3226)*(X3227=5) +
344 MAX(0,X3221)*(X3222=1|X3222=6) +
345 MAX(0,X3329) + MAX(0,X3324) - MAX(0,X3326)*(X3327=5) +
346 MAX(0,X3321)*(X3322=1|X3322=6) +
347 MAX(0,X3335) + (X507/10000)*(X513+X526)+
348 MAX(0,X3408)+MAX(0,X3412)+MAX(0,X3416)+MAX(0,X3420)+
349 MAX(0,X3424)+MAX(0,X3428);

```

```

350
351 * other nonfinancial ;
352 OTHNFIN = X4022 + X4026 + X4030 -
353 X4022*(X4020 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77)) -
354 X4026*(X4024 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77)) -
355 X4030*(X4028 IN (61, 62, 63, 64, 65, 66, 72, 73, 74, 77));
356
357 NFIN=VEHIC+HOUSES+ORESRE+NNRESRE+BUS+OTHNFIN;
358
359 NHNFIN=NFIN-HOUSES;
360
361 ASSET=FIN+NFIN;
362
363 ***** Debts *****;
364
365 * housing debt (mortgage, home equity loans and HELOCs);
366 * spillover LOCs divided between HE and other;
367 IF (X1108+X1119+X1130)>=1 THEN DO;
368 MRTHEL=X805+X905+X1005+
369 X1108*(X1103=1)+X1119*(X1114=1)+X1130*(X1125=1)+
370 MAX(0, X1136)*(X1108*(X1103=1)+
371 X1119*(X1114=1)+X1130*(X1125=1))/(X1108+X1119+X1130);
372 END;
373 ELSE DO;
374 MRTHEL=X805+X905+X1005+. 5*(MAX(0, X1136));
375 END;
376
377 * other lines of credit ;
378 IF (X1108+X1119+X1130)>=1 THEN DO;
379 OTHLOC=X1108*(X1103^=1)+X1119*(X1114^=1)+X1130*(X1125^=1)+
380 MAX(0, X1136)*(X1108*(X1103^=1)+X1119*(X1114^=1)+
381 X1130*(X1125^=1))/(X1108+X1119+X1130);
382 END;
383 ELSE DO;
384 OTHLOC=. 5*(MAX(0, X1136));
385 END;
386
387 * debt for other residential property => land contracts, Section E
388 residential, misc vacation, and Section I cottage/vacation home (67);
389 * NB debt for nonresidential real estate is netted out of assets;
390 ORESDBT=X1417 + X1517 + X1617 + X1621
391 + (X1703 IN (12, 14, 21, 22, 25, 40, 41, 42, 43, 44, 49, 50, 52, 999)) *
392 X1715*(X1705/10000)
393 + (X1803 IN (12, 14, 21, 22, 25, 40, 41, 42, 43, 44, 49, 50, 52, 999)) *
394 X1815*(X1805/10000)
395 + (X1903 IN (12, 14, 21, 22, 25, 40, 41, 42, 43, 44, 49, 50, 52, 999)) *
396 X1915*(X1905/10000) + X2006
397 + X2723*(X2710 IN (67)) + X2740*(X2727 IN (67))
398 + X2823*(X2810 IN (67)) + X2840*(X2827 IN (67))
399 + X2923*(X2910 IN (67)) + X2940*(X2927 IN (67));
400
401 * credit card debt;
402 CCBAL = MAX(0, X427)+MAX(0, X413)+MAX(0, X421)+MAX(0, X430)+
403 MAX(0, X424)+MAX(0, X7575);
404
405 * installment loans;
406 * '95 has additional car loan, X7169, and new loan mopups, X7179 and X7183;
407 INSTALL=X2218+X2318+X2418+X7169+X2424+
408 X2519+X2619+X2625+
409 X2723*(X2710^=67)+X2740*(X2727^=67)+
410 X2823*(X2810^=67)+X2840*(X2827^=67)+
411 X2923*(X2910^=67)+X2940*(X2927^=67)+X7183+
412 X7824+X7847+X7870+X7924+
413 X7947+X7970+X7179+
414 X1044+X1215+X1219;
415
416 * other debts (loans vs pensions, loans vs life insurance,
417 margin loans, miscellaneous);
418 ODEBT=MAX(0, X4229)*(X4230=5)+MAX(0, X4329)*(X4330=5)+MAX(0, X4429)*(X4430=5)+
419 MAX(0, X4829)*(X4830=5)+MAX(0, X4929)*(X4930=5)+MAX(0, X5029)*(X5030=5)+
420 MAX(0, X4010)+MAX(0, X4032)+MAX(0, X3932);
421
422 * total debt;
423 DEBT=MRTHEL+ORESDBT+OTHLOC+CCBAL+INSTALL+ODEBT;
424
425 * total net worth;
426 NETW=ASSET-DEBT;
427
428 mortgage = max(0, x805) + max(0, x905) + max(0, x1005);

```

Define_Variab les.txt Monday, February 06, 2006 11:16:24 AM
429 homeq = mrthel - mortgage;
430
431 /* DEMOGRAPHICS */
432
433 ageh=x14; /*Husband's age (check not respondent's age) */
434 Ages=x19; /*Spouse's age*/
435 Edh=x5901;
436 totinc = x5729;
437
438 run;

NOTE: The data set MYDISK.SMALL2001NW has 22210 observations and 590 variables.
NOTE: DATA statement used:
real time 1:17.41
cpu time 6.15 seconds

439 * program SCF01weights.SAS;
440 * Reads in the sampling variation;
441 * Karl's program (based on Karen Pence's program), 5/16/03;
442

443 libname mydisk 'd:\scfs\2001';
NOTE: Libref MYDISK was successfully assigned as follows:
Engine: V7
Physical Name: d:\scfs\2001

444 options obs=max;
445

446
447 proc import infile='d:\scfs\2001\scf2001rw1' data='d:\scfs\2001\sampleweight2001';
WARNING: The Base Product product with which CIMPORT is associated will expire within 30 days.
Please contact your SAS installation representative to have it renewed.

448

NOTE: Proc CIMPORT begins to create/update data set MYDISK.SAMPLEWEIGHT2001
NOTE: Data set contains 2000 variables and 4442 observations.
Logical record length is 12008

NOTE: PROCEDURE CIMPORT used:
real time 52.06 seconds
cpu time 8.88 seconds

449 data mydisk.secondhalf2001;

WARNING: The Base Product product with which DATASTEP is associated will expire within 30 days.
Please contact your SAS installation representative to have it renewed.

450 set mydisk.sampleweight2001;
451 drop MM1-MM999;

452

453 run;

NOTE: The data set MYDISK.SECONDHALF2001 has 4442 observations and 1001 variables.
NOTE: DATA statement used:
real time 26.46 seconds
cpu time 1.70 seconds