THIS CODE CAN BEGIN WHERE YESTERDAY’S CODE ENDS. THE SAME “PROFILES” FILE THAT IS LOADED IN THE PREVIOUS LAB EXERCISE IS NEEDED FOR THIS ONE.

local hhproduction clean laund cook hhmaint lawngar hhmgmt petcare purch chcarehh chcarenhh sechh secnhh elcarehh elcarenhh elcarev trav

use profiles, clear

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Want to use four different smoothers and compare the results

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

First is "autosmoo". This command is not part of the standard Stata package. It is an .ado file that can be accessed from the web and installed and then it functions just like a regular stata command. In general, .ado files are written for new experimental methods, to give the software engineers time to try things out before making something part of the standard package.

“autosmoo” is a cross-validation smoother that allows weights

Need to use “findit autosmoo” then read what comes up there and click on “install.”

\*/

findit autosmoo

foreach sex in 1 2 {

foreach vvv in `hhproduction' paid workrel {

if ("`vvv'"=="chcarehh"|"`vvv'"=="chcarenhh"|"`vvv'"=="sechh"|"`vvv'"=="secnhh"|"`vvv'"=="elcarehh"|"`vvv'"=="elcarenhh"|"`vvv'"=="elcarev"|"`vvv'"=="hhmaint"|"`vvv'"=="hhmgmt") {

autosmoo `vvv'`sex' [w=totwt`sex'], gen (`vvv'`sex'\_sm3) kmin(1) kmax(3) nograph

}

if ("`vvv'"~="chcarehh"&"`vvv'"~="chcarenhh"&"`vvv'"~="sechh"&"`vvv'"~="secnhh"&"`vvv'"~="elcarehh"&"`vvv'"~="elcarenhh"&"`vvv'"~="elcarev"&"`vvv'"~="hhmaint"&"`vvv'"~="hhmgmt") {

autosmoo `vvv'`sex' [w=totwt`sex'], gen (`vvv'`sex'\_sm3) nograph

}

}

}

save profiles, replace

foreach sex in 1 2 {

foreach vvv in `hhproduction' paid workrel {

if ("`vvv'"=="chcarehh"|"`vvv'"=="chcarenhh"|"`vvv'"=="sechh"|"`vvv'"=="secnhh"|"`vvv'"=="elcarehh"|"`vvv'"=="elcarenhh"|"`vvv'"=="elcarev"|"`vvv'"=="hhmaint"|"`vvv'"=="hhmgmt") {

lowess `vvv'`sex' age, gen (`vvv'`sex'\_sm1) bwidth(0.1) nograph

}

if ("`vvv'"~="chcarehh"&"`vvv'"~="chcarenhh"&"`vvv'"~="sechh"&"`vvv'"~="secnhh"&"`vvv'"~="elcarehh"&"`vvv'"~="elcarenhh"&"`vvv'"~="elcarev"&"`vvv'"~="hhmaint"&"`vvv'"~="hhmgmt") {

lowess `vvv'`sex' age, gen (`vvv'`sex'\_sm1) bwidth(0.6) nograph

}

}

}

save profiles, replace

capture rm tudata.csv

foreach sex in 1 2 {

foreach vvv in `hhproduction' paid workrel {

use profiles, clear

gen spanval="cv"

if ("`vvv'"=="chcarehh"|"`vvv'"=="chcarenhh"|"`vvv'"=="sechh"|"`vvv'"=="secnhh"|"`vvv'"=="elcarehh"|"`vvv'"=="elcarenhh"|"`vvv'"=="elcarev"|"`vvv'"=="hhmaint"|"`vvv'"=="hhmgmt") {

drop spanval

gen spanval=.05

}

keep age totwt`sex' `vvv'`sex' spanval

order age totwt`sex' `vvv'`sex' spanval

outsheet using tudata.csv, comma

shell R --vanilla < supsmu\_timeuse.r > supsmu.out

insheet using tusmoothed.csv, comma names clear

rename x age

rename y `vvv'`sex'\_sm4

keep age `vvv'`sex'

sort age

save `vvv'`sex'sm4, replace

use profiles, clear

sort age

merge age using `vvv'`sex'sm4

drop \_merge

sort age

save profiles, replace

rm `vvv'`sex'sm4.dta

rm tudata.csv

rm tusmoothed.csv

}

}

save profiles, replace

foreach sex in 1 2 {

use profiles, clear

gen truncwt`sex'=round(totwt`sex'/5000000)

keep age clean`sex' laund`sex' cook`sex' hhmaint`sex' lawngar`sex' hhmgmt`sex' petcare`sex' purch`sex' chcarehh`sex' chcarenhh`sex' sechh`sex' secnhh`sex' elcarehh`sex' elcarenhh`sex' elcarev`sex' trav`sex' paid`sex' workrel`sex' truncwt`sex'

expandcl truncwt`sex', cluster(age) gen(wid)

foreach vvv in `hhproduction' paid workrel {

if ("`vvv'"=="chcarehh"|"`vvv'"=="chcarenhh"|"`vvv'"=="sechh"|"`vvv'"=="secnhh"|"`vvv'"=="elcarehh"|"`vvv'"=="elcarenhh"|"`vvv'"=="elcarev"|"`vvv'"=="hhmaint"|"`vvv'"=="hhmgmt") {

lowess `vvv'`sex' age, gen (`vvv'`sex'\_sm2) bwidth(0.1) nograph

}

if ("`vvv'"~="chcarehh"&"`vvv'"~="chcarenhh"&"`vvv'"~="sechh"&"`vvv'"~="secnhh"&"`vvv'"~="elcarehh"&"`vvv'"~="elcarenhh"&"`vvv'"~="elcarev"&"`vvv'"~="hhmaint"&"`vvv'"~="hhmgmt") {

lowess `vvv'`sex' age, gen (`vvv'`sex'\_sm2) bwidth(0.6) nograph

}

display "`vvv'`sex'"

}

keep age \*\_sm2

collapse (mean) \*\_sm2, by(age)

sort age

save `sex'\_sm2results, replace

}

use profiles, clear

sort age

merge age using 1\_sm2results

drop \_merge

sort age

merge age using 2\_sm2results

drop \_merge

save profiles, replace

rm 1\_sm2results.dta

rm 2\_sm2results.dta

foreach smoothertype in 1 2 3 4 {

foreach sex in 1 2 {

egen tothha`sex'\_sm`smoothertype'=rowtotal(clean`sex'\_sm`smoothertype' laund`sex'\_sm`smoothertype' cook`sex'\_sm`smoothertype' hhmaint`sex'\_sm`smoothertype' lawngar`sex'\_sm`smoothertype' hhmgmt`sex'\_sm`smoothertype' petcare`sex'\_sm`smoothertype' purch`sex'\_sm`smoothertype' chcarehh`sex'\_sm`smoothertype' chcarenhh`sex'\_sm`smoothertype' sechh`sex'\_sm`smoothertype' secnhh`sex'\_sm`smoothertype' elcarehh`sex'\_sm`smoothertype' elcarenhh`sex'\_sm`smoothertype' elcarev`sex'\_sm`smoothertype' trav`sex'\_sm`smoothertype')

egen totwk`sex'\_sm`smoothertype'=rowtotal(paid`sex'\_sm`smoothertype' workrel`sex'\_sm`smoothertype')

}

}

/\* Graphs to check smoothing (commented out after first run through) \*/

foreach sex in 1 2 {

foreach vvv in `hhproduction' paid workrel tothha totwk {

graph twoway (line `vvv'`sex'\_sm1 age) (line `vvv'`sex'\_sm2 age) (line `vvv'`sex'\_sm3 age) (line `vvv'`sex'\_sm4 age) (scatter `vvv'`sex' age [w=totwt`sex'], msymbol(circle\_hollow)), xtitle("Age") ytitle("Avg Hrs Per Day") legend(label(1 unwtd lowess) label(2 wtd lowess) label(3 wtd autosmoo) label(4 R wtd supsmu)) title (`vvv' `sex')

graph export review\_graphs/smoothcheck\_`vvv'`sex'sec.png, replace

}

}