

Physical Activity, METMIN and EXGRP, in 1989-95 Cohort

The 1989-95 cohort data uses a different method from other ALSWH cohorts to calculate the physical activity variables. This follows from work by Brown et al (2012).

The original ALSWH cohorts use the physical activity variables called Exercise Statistic, EXSTAT, and Exercise Statistic Grouped, ExGrp. The 1989-95 cohort uses similar variables, METMIN and EXGRP, but calculated differently.

Survey 1 of the 1989-95 cohort asked the following questions about physical activity. Subsequent surveys after Survey 1 have continued this.

The next two questions are about the amount of physical activity you did LAST WEEK.

<p>Please state how many times you did each type of activity <u>last week</u>.</p> <p><i>Only count activities that lasted for 10 minutes or more.</i></p> <p><i>(If you did not do an activity, please type "0")</i></p>	
Walking briskly (<i>for recreation or exercise, or to get from place to place</i>)	EXER004
Moderate leisure activity (<i>like social tennis, moderate exercise classes, recreational swimming, dancing</i>)	EXER005
Vigorous leisure activity (<i>that makes you breathe harder or puff and pant like aerobics, competitive sport, vigorous cycling, running, swimming</i>)	EXER006
Vigorous household or garden chores (<i>that make you breathe harder or puff and pant</i>)	EXER011

<p>Please state how much time you spent altogether doing each type of activity <u>last week</u>.</p> <p><i>Add up all the times you spent in each activity to get the total time for each</i></p>		
Walking briskly (<i>for recreation or exercise, or to get from place to place</i>)	EXERHRS1	EXERMINS1
Moderate leisure activity (<i>like social tennis, moderate exercise classes, recreational swimming, dancing</i>)	EXERHRS2	EXERMINS2
Vigorous leisure activity (<i>that makes you breathe harder or puff and pant like aerobics, competitive sport, vigorous cycling, running, swimming</i>)	EXERHRS3	EXERMINS3
Vigorous household or garden chores (<i>that make you breathe</i>	EXERHRS4	EXERMINS4

<i>harder or puff and pant)</i>		
---------------------------------	--	--

A new method for calculating the met minutes (metmins) and the exercise status groups (exgrp) was used for this cohort's data. The critical code is this (the full code is at the base of this document):

```
/* alswh */
metmin=sum((totmina*3.33),(totminb*3.33),(totminc*6.66));
/*if totmin>2400 then metmin=.;*/
if 0<=metmin<33.3 then exgrp=1;
else if 33.3<=metmin<500 then exgrp=2;
else if 500<=metmin <1000 then exgrp=3;
else if metmin>=1000 then exgrp=4;
```

References

Brown, W., Bauman, A., Bull, F. et al, Development of evidence-based physical activity recommendations for adults (18–64 years). Australian Government Department of Health, ; 2012.

Appendix Full Code

The SAS code for deriving the metmins and exgrp variables

```
/* physical activity */

array leisure{3}  exer004  exer005  exer006  ;
array numtimes{4} exer004  exer005  exer006  exer011  ;

array hours{4}    exerhrs1  exerhrs2  exerhrs3  exerhrs4  ;
array minutes{4}  exermins1  exermins2  exermins3  exermins4  ;

array totmins{3}  totmina  totminb  totminc  ;

/*****
# times doing leisure activities

If some items in this group have a nonzero response AND
no items in this group have a response of 'zero'
*****/
```

```

    THEN code missing values to zero.
*****
    if nmiss (of numtimes{*}) in (0,4) then ;
    else if exer004=0 or exer005=0 or exer006=0 or exer011=0 then ;
    else do i = 1 to 4 ;
        if numtimes{i}=. then numtimes{i}=0 ;
    end ;

/*****
    hours & minutes spent doing leisure activities

    If hours spent doing activities is not missing AND
    the corresponding minutes are missing
    THEN code missing values in minutes to 0.

    (Similarly, for minutes not missing and hours missing)
*****
    do i = 1 to 4 ;
        if hours{i} ne . and minutes{i} = . then minutes{i} = 0 ;
        if hours{i} = . and minutes{i} ne . then hours{i} = 0 ;
    end ;

/*****
***
*** If number of times=0 and hours=. and mins=. ***
*** then set hours and mins to 0. ***
*** ***
*** If hours=0 and mins=0 and number of times=. ***
*** then set number of times to 0. ***
*** ***
*****/
do a=1 to 4 ;
    if numtimes(a)=0 and hours(a)=. and minutes(a)=. then do ;
        hours(a)=0 ;
        minutes(a)=0 ;
    end ;
end ;

do a=1 to 4 ;
    if numtimes(a)=. and hours(a)=0 and minutes(a)=0 then numtimes(a)=0 ;
end ;

/*****
***
*** If hours>0 and mins>0 and number of times=. ***
*** then set number of times to 0. ***
*** ***
*****/
    do a=1 to 4 ;
        if numtimes(a)=. and 0<=hours(a)<=99 and 0<=minutes(a)<=90
        then numtimes(a)=0 ;
    end ;

/*****
    Calculate total # times leisure activity reported
*****/
    exertot = exer004 + exer005 + exer006 ;

```

```

/*****
  Re-assign extreme values for the # times reported doing
  leisure activities, that is totals > 56 times
  *****/
if exertot >56 then do i = 1 to 3 ;
  leisure{i} = round ( ((56/exertot) * leisure{i}), 1) ;
end ;

else if exertot = . then do ;
  if (exer004+exer005) >56 then do ;
    exer004 = round ( ( (56/(exer004+exer005) ) * exer004), 1) ;
    exer005 = round ( ( (56/(exer004+exer005) ) * exer005), 1) ;
  end ;
  else if (exer004+exer006) > 56 then do ;
    exer004 = round ( ( (56/(exer004+exer006) ) * exer004), 1) ;
    exer006 = round ( ( (56/(exer004+exer006) ) * exer006), 1) ;
  end ;
  else if (exer005+exer006) >56 then do ;
    exer005 = round ( ( (56/(exer005+exer006) ) * exer005), 1) ;
    exer006 = round ( ( (56/(exer005+exer006) ) * exer006), 1) ;
  end ;
  else do i = 1 to 3 ;
    if leisure{i} >56 then leisure{i}=56 ;
  end ;
end ;

if exer011>56 then exer011=56 ;

/*****
  Calculate the total minutes doing each leisure activity
  *****/
do i = 1 to 3 ;
  totmins{i} = (60*hours{i}) + minutes{i} ;
end ;

/*****
  Calculate total minutes of leisure activity reported
  *****/
exermintot = round ((totmina + totminb + totminc), .01);

/*****
  Re-assign extreme values for the total time reported doing
  leisure activities, that is totals > 40 hrs (>2400 minutes)
  *****/
if exermintot >2400 then do i = 1 to 3 ;
  totmins{i} = round ( ((2400/exermintot) * totmins{i}), .01) ;
end ;

else if exermintot = . then do ;
  if (totmina + totminb) >2400 then do ;
    totmina = round(((2400/(totmina + totminb) ) * totmina), .01) ;
    totminb = round(((2400/(totmina + totminb) ) * totminb), .01) ;
  end ;
  else if (totmina + totminc) > 2400 then do ;
    totmina = round(((2400/(totmina + totminc) ) * totmina), .01) ;
    totminc = round(((2400/(totmina + totminc) ) * totminc), .01) ;
  end ;
end ;

```

```

        end ;
    else if (totminb + totminc) >2400 then do ;
        totminb = round(((2400/(totminb + totminc) ) * totminb), .01) ;
        totminc = round(((2400/(totminb + totminc) ) * totminc), .01) ;
        end ;
    else do i = 1 to 3 ;
        if totmins{i} >2400 then totmins{i}=2400 ;
        end ;
    end ;
/* AHS exgrp */
metmin=sum((totmina*3.5),(totminb*5),(totminc*7.5));
/*if totmin>2400 then metmin=.;*/
if 0<=metmin<50 then exgrp=1;
else if 50<=metmin<800 then exgrp=2;
else if 800<=metmin<1600 or (metmin>=1600 and totminc<120) then exgrp=3;
else if metmin>=1600 then exgrp=4;
/* AHS activity */
actmin=sum(totmina,totminb,(totminc*2));
actnum=sum(exer004,exer005,exer006);
activity=.;
if actmin=0 then activity=1;
else if 0<actmin<150 or (actnum<5 and actmin>=150) then activity=2;
else if actmin>=150 then activity=3;
/* alsw new */
metmin=sum((totmina*3.33),(totminb*3.33),(totminc*6.66));
/*if totmin>2400 then metminnew=.;*/
if 0<=metmin<33.3 then exgrp=1;
else if 33.3<=metmin<500 then exgrp=2;
else if 500<=metmin <1000 then exgrp=3;
else if metmin>=1000 then exgrp=4;
label exgrp='ALSWH exercise groups';

```