

---

Subject: Problem in converting birth date CMC from Afghan calendar to Gregorian calendar

Posted by [jing](#) on Thu, 23 Feb 2017 03:39:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

HI,

I am using the data for Afghanistan 2015. I got a problem in converting the birth date into Gregorian calendar. It is the Afghan calendar used in the survey. As far as I understand, the Afghan new year starts from around 21st of March (varies in leap year), so information of day is needed if we want to convert Afghan calendar into Gregorian calendar. But only month and year of birth are asked.

In Afghanistan 2010, it is the same that only month and year of birth are asked. But both birth date in Afghan calendar and Gregorian calendar are available (Q102MG, Q102MG, Q102CG). I checked the questionnaire, only one calendar was used (probably Afghan calendar), so the month and year of birth must be converted from the Afghan calendar, May I ask how it was done? I think I probably need to do the same for year 2015 in order to get consistent estimation.

thanks

Jing

---

#### File Attachments

1) [Capture.PNG](#), downloaded 602 times

---

---

Subject: Re: Problem in converting birth date CMC from Afghan calendar to Gregorian calendar

Posted by [Trevor-DHS](#) on Fri, 24 Feb 2017 06:42:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi Jing,

Below is the conversion function used in CSPro:

```
{ Convert Persian calendar to Gregorian }
function dconvert(p_day,p_month,p_year); { Assume for all years that the vernal equinox is at
March 21st }
  xtemp = 1;
  if validyr(p_year) & p_month in 1:12 then

    g_year = p_year + 621;                                { 1;Hammal - 31 days (March-April)
    }
    if !(p_day in 1:31) then                                { 2;Saur - 31 days (April-May)
    }
    p_day = 15;                                             { 3;Jauza - 31 days (May-June)
    }
  endif;                                                    { 4;Saratan - 31 days (June-July)
```

```

}
{ number of days since start Persian year }
(July-August) } { 5;Asad - 31 days
    box p_month => days;
(August-September) } { 6;Sonbola - 31 days
    1-6 => (p_month-1) * 31 + p_day;
- October) } { 7;Mizan - 30 days (September
    => 186 + (p_month-7) * 30 + p_day;
(October-November) } { 8;Aqrab - 30 days
    endbox;
December) } { 9;Qaus - 30 days (Novemembr -
    { Days between vernal equinox and Jan. 1st }
(December - January) } { 10;Jadi - 30 days
    equinox = 80;
    }
    { number of days Gregorian is under way }
(February-March) } { 11;Dalw - 30 days (January-February)
    tdays = equinox + days;
    leapyear = (g_year%4 = 0);
    if !leapyear then
        if tdays > 365 then
            tdays = tdays - 365;
            g_year = g_year + 1;
        endif;
    else
        if tdays > 366 then
            tdays = tdays - 366;
            g_year = g_year + 1;
        endif;
    endif;
    if leapyear & tdays >= 60 then tdays = tdays - 1 endif;

{ Determine Gregorian month }
box tdays => g_month;
    <= 31 => 1;
    <= 59 => 2;
    <= 90 => 3;
    <= 120 => 4;
    <= 151 => 5;
    <= 181 => 6;
    <= 212 => 7;
    <= 243 => 8;
    <= 273 => 9;
    <= 304 => 10;
    <= 334 => 11;
    => 12;
endbox;

```

```

{ And the day in the month }
box      tdays => g_day;
    <= 31 => tdays;
    <= 59 => tdays - 31;
    <= 90 => tdays - 59;
    <= 120 => tdays - 90;
    <= 151 => tdays - 120;
    <= 181 => tdays - 151;
    <= 212 => tdays - 181;
    <= 243 => tdays - 212;
    <= 273 => tdays - 243;
    <= 304 => tdays - 273;
    <= 334 => tdays - 304;
    => tdays - 334;
endbox;
else
    g_day = p_day;          { Cannot determine }
    if validyr(p_year) then
        g_year = p_year + 621;    { most likely; first 9 months }
        g_month = p_month;        { DK/missing }
    else
        if p_month in 1:12 then
            g_year = p_year;      { DK/missing }
            box p_month => g_month;
                1-9 => p_month + 3; { most likely; 21 days out 31 }
                => p_month - 9;
            endbox;
        else
            g_year = p_year;    { DK/missing }
            g_month = p_month;  { DK/missing }
        endif;
    endif;
endif;
if g_year = 2011 then
    g_year = 2010;
    g_month = 12;
endif;

dconvert = xtemp;
{errmsg("In:%d-%d-%d and out: %d-%d-%d",p_day,p_month,p_year,g_day,g_month,g_year); }
end;

```

---

Subject: Re: Problem in converting birth date CMC from Afghan calendar to  
Gregorian calendar

Posted by [jing](#) on Fri, 24 Feb 2017 07:21:34 GMT

Hi,

Thanks Trevor for the codes.

best

Jing

---

---

Subject: Re: Problem in converting birth date CMC from Afghan calendar to  
Gregorian calendar

Posted by [tomtom\\_m](#) on Tue, 14 Mar 2017 10:20:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

All, here is some STATA code that does the same thing. (Substantively debugged, remaining errors are still my fault!)

We impute days of birth for mothers and children which is also required for more precise estimation of child mortality, and then create Gregorian CMCs.

**\*\* STATA CODE TO TRANSLATE PERSIAN TO GREGORIAN CALENDAR\*\***

**\*\*BIRTH DATES\*\***

**\*MOTHERS\***

```
gen v010g= v010+621 /* Gregorian year at 21 March of Year of Birth*/
gen leapyear = mod(v010g+1,4)==0 /*is the February of the following year a leap year? */
gen v009day = runiformint(1,31) if inrange(v009,1,6)
replace v009day=runiformint(1,30) if inrange(v009,7,11)
replace v009day=runiformint(1,29+leapyear) if v009==12

gen momdays= (v009-1)*31 + v009day if inrange(v009,1,6) /*momdays= days from 21 Mar*/
replace momdays=186+(v009-7)*30+v009day if inrange(v009,7,12)

gen DoB = mdy(3,21,v010g)+momdays
format DoB %d
gen v009g=month(DoB)
gen v011g = (year(DoB)-1900)*12+v009g
move v009g v009
move v010g v010
move v011g v011
move DoB v011
la var v009g "Gregorian month of birth"
la var v010g "Gregorian year of birth"
la var v011g "Gregorian CMC of birth"
la var DoB "Gregorian DoB - imputed"
drop leapyear momdays v009day
```

**\* CHILDREN\***

```

local border 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20
foreach item of local border {
gen b2g_`item' = b2_`item'+621 /* Gregorian year at 21 March of Year of Birth*/
gen leapyear_`item' = mod(b2g_`item'+1,4)==0
gen b1_`item'day = runiformint(1,31) if inrange(b1_`item',1,6)
replace b1_`item'day=runiformint(1,30) if inrange(b1_`item',7,11)
replace b1_`item'day=runiformint(1,29+leapyear_`item') if b1_`item'==12

gen kiddays_`item' = (b1_`item'-1)*31 + b1_`item'day if inrange(b1_`item',1,6) /*kiddays= days
from 21 Mar*/
replace kiddays_`item'=186+(b1_`item'-7)*30+b1_`item'day if inrange(b1_`item',7,12)

gen DoB_`item' = mdy(3,21,b2g_`item')+kiddays_`item'
format DoB_`item' %d
gen b1g_`item'=month(DoB_`item')
replace b2g_`item'=year(DoB_`item')
gen b3g_`item' = (b2g_`item'-1900)*12+b1g_`item'
drop kiddays_`item'
drop leapyear_`item'
drop b1_`item'day
move b1g_`item' b1_`item'
move b2g_`item' b2_`item'
move b3g_`item' b3_`item'
move DoB_`item' b3g_`item'
la var b1g_`item' "Gregorian month of birth"
la var b2g_`item' "Gregorian year of birth"
la var b3g_`item' "Gregorian CMC of birth"
la var DoB_`item' "Gregorian DoB - imputed"
}

```

**\*\*INTERVIEW DATES\*\***

```

gen v007g = v007+621 /* Gregorian year at 21 March of Year of Birth*/
gen intdays = (v006-1)*31 + v016 if inrange(v006,1,6) /*momdays= days from 21 Mar*/
replace intdays=186+(v006-7)*30+v016 if inrange(v006,7,12)
gen Dol = mdy(3,21,v007g)+intdays
format Dol %d
gen v006g=month(Dol)
gen v008g = (year(Dol)-1900)*12+v006g
drop intdays
move v006g v006
move v007g v007
move v008g v008
move Dol v008
la var v006g "Gregorian month of interview"
la var v007g "Gregorian year of interview"
la var v008g "Gregorian CMC of interview"

```

la var Dol "Gregorian date of interview"

**\*\*SWAP GREGORIAN AND PERSION DATES\*\***

```
rename v006 v006p
rename v006g v006
rename v007 v007p
rename v007g v007
rename v008 v008p
rename v008g v008
rename v009 v009p
rename v009g v009
rename v010 v010p
rename v010g v010
rename v011 v011p
rename v011g v011
rename b1_((##)) b1p_((##))
rename b2_((##)) b2p_((##))
rename b3_((##)) b3p_((##))
rename b1g_((##)) b1_((##))
rename b2g_((##)) b2_((##))
rename b3g_((##)) b3_((##))
```

---