

---

Subject: roportion of existing nets that have been used the previous night

Posted by [Nelly\\_WHO](#) on Wed, 03 May 2017 12:37:43 GMT

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

---

Hi Cameron,

I am trying to match DHS stat compiler data on the Proportion of existing nets that have been used the previous night. Am I using the correct variables? hml10 and hml21?

Thank your very much,

Below is an extract of my code.

Melly

```
// Whether each bed net owned by the household is an ITN
```

```
cap confirm variable hml10_01
if _rc == 0 {
  forvalues n = 1/9 {
    cap rename hml10_0`n' bednet_is_itn`n'
  }
  forvalues n = 10/100 {
    cap rename hml10_`n' bednet_is_itn`n'
  }
}
cap confirm variable hml10_1
if _rc == 0 {
  forvalues n = 1/9 {
    cap rename hml10_`n' bednet_is_itn`n'
  }
  forvalues n = 10/100 {
    cap rename hml10_`n' bednet_is_itn`n'
  }
}
```

```
// Determine whether the survey has the ITN variable
```

```
cap confirm variable bednet_is_itn1
if _rc == 0 {
  summ bednet_is_itn1
  if `r(N)' != 0 {
```

```
  // number of ITNs
```

```
  local ITNvar bednet_is_itn*
  egen nbITN = rowtotal(`ITNvar')
```

```
cap confirm variable net_slept_under1
```

```
if _rc != 0 {
  cap confirm variable hml21_01
  if _rc == 0 {
    forvalues n = 1/9 {
      rename hml21_0`n' net_slept_under`n'
```

```

}
forvalues n = 10/100 {
  cap rename hml21_`n' net_slept_under`n'
}
}
cap confirm variable hml21_1
if _rc == 0 {
  forvalues n = 1/9 {
    cap rename hml21_`n' net_slept_under`n'
  }
  forvalues n = 10/100 {
    cap rename hml21_`n' net_slept_under`n'
  }
}
}
forvalues n = 1/100 {
  cap replace net_slept_under`n' = . if net_slept_under`n' > 1
}
}

// make a variable for how many ITNs were slept under the night before
forvalues n = 1/100 {
  cap gen itn_slept_under`n' = 1 if net_slept_under`n' == 1 & bednet_is_itn`n' == 1
  cap replace itn_slept_under`n' = 0 if net_slept_under`n' == 0 | bednet_is_itn`n' == 0
}

// calculate the total number of ITNs that were slept under
egen total_itns_slept_under = rowtotal(itn_slept_under*)
// calculate the fraction of available nets that were used the night before
gen frac_nets_used = total_itns_slept_under / nbITN

// loop through to calculate survey-weighted means & SEs
svyset [pweight=sample_weight], psu(cluster_n)
local m frac_nets_used

// TOTAL
svy: mean `m'
ereturn list
matrix mean_matrix = e(b)
matrix variance_matrix = e(V)
local mean = mean_matrix[1,1]
local se = sqrt(variance_matrix[1,1])
gen prop_`m' = `mean'
gen seprop_`m' = `se'
gen uci_`m' = prop_`m' + 1.96*seprop_`m'
gen lci_`m' = prop_`m' - 1.96*seprop_`m'
drop seprop*

```