# Tips och Tricks för datahantering i Stata

Christel Häggström

Registercentrum Norr, Region Västerbotten &

Inst för folkhälsa och klinisk medicin, Umeå Universitet

Registercentrum Norr

Kliniska Studier Sverige Forum Norr

UMEÅ UNIVERSITY

# Agenda datahantering i Stata

- Flera-till-1 rad per individ
- Datumhantering
- Samkörning av filer
- Hantering av strängvariabler
- Spara och gå tillbaka till datat
- ”Loopande” kommandon
- Hantering av ÅÖÄ
- Smidig baslinjetabell

Med exempel från registerdata…

# Flera rader av individer → en rad per individ

- Kolla först om det finns några rader i datat med identiska data, om det finns så ta bort dem

*duplicates drop*

- Kolla på hur många det finns med flera rader, kolla på dessa i detalj

*by lopnr : gen dub= _n*
*tab dub*

- Du väljer dubletter med tidigaste datum (av anydate)

*sort lopnr anydate*
*by lopnr : gen dub= _n*
*tab dub*
*list lopnr if dub==2 /*kolla på dessa lopnr närmare för att dubbelkolla att sorteringen map anydate ser bra ut */*
*drop if dub==2*

# Flera rader av individer → en rad per individ

- Formatera om data mellan “wide” format (en observation/rad för en individ) och “long” (flera observations/rader för en individ)
- Vanligt när det finns upprepade rader per individ från tex patient- eller läkemedelsregistret.

*reshape wide*

*reshape long*

*long*

| *i* | *j* | *stub* |
|---|---|---|
| 1 | 1 | 4.1 |
| 1 | 2 | 4.5 |
| 2 | 1 | 3.3 |
| 2 | 2 | 3.0 |

reshape ⟷

*wide*

| *i* | *stub***1** | *stub***2** |
|---|---|---|
| 1 | 4.1 | 4.5 |
| 2 | 3.3 | 3.0 |

# Datumhantering

- Datum kan ofta komma i annat format än datumformat, gör om till datumformat

*format anydate %d*

- Du vill lägga in ett datum som variabel

*gen end_FU=td(31dec2019)*
*format end_FU %d*

- Det går bra att använda andra kommandon för datum, tex

*codebook anydate*
*gen Last_day_of_FU=min(death_date,end_FU, emig_date)*
*gen FU_time=(Last_day_of_FU - First_day_of_FU)/365.25*
*sum FU_time*

- Ibland kan dag eller månad vara kodat som 00, tex från dödsorsaksregistret.
- Ett sätt är att använda strängvariabler för att kolla i detalj på hur mycket ”00” det finns i datat. Formatet här är ”XXXXYYZZ”

```
tostring anydate, replace
gen anydate_string=strtrim(anydate)
gen year=substr(anydate_string,1,4)
gen month=substr(anydate_string,5,2)
gen day=substr(anydate_string,7,2)

tab year, missing  /* Check how many ”00” in these variables */
tab month, missing
tab day, missing

replace day="30" if month=="00" & day=="00"
replace month="06" if month=="00" & day=="30"
replace day="15" if day=="00"

egen anydate_string_mod=concat(day month year)
gen anydate_mod =date(anydate_string_mod, "DMY")
format anydate_mod %d
codebook anydate_mod  /*double check that the created date looks OK */(
drop year month day
```

# Samkörning av filer - merge

För att lägga till mer information/kolumner på existerande individer i datat.

Merge 1:1 kräver en unik rad per individ (lopnr)

Keepusing smidigt om man inte vill ha alla variabler

- *merge 1:1 lopnr*
- *merge 1:n lopnr*
- *merge n:n lopnr*
- *merge 1:1 lopnr using "migrationer.dta", keepusing (utv_date)*

# Samkörning av filer - append

För att lägga till mer individer/rader i datat

- *append*

# Kategorisera variabler i kategorier

- Kommandot ”egen” innehåller flera användbara alternativ

*egen dia_year_cat=cut(dia_year), at(1996, 2005, 2013, 2020)  icodes label*

Generate `newv1` for distinct groups of `v1` and `v2`, and create and apply value label `mylabel`

```
egen newv1 = group(v1 v2), label(mylabel)
```

Generate `newv2` equal to the minimum of `v1`, `v2`, and `v3` for each observation

```
egen newv2 = rowmin(v1 v2 v3)
```

Generate `newv3` equal to the overall sum of `v1`

```
egen newv3 = total(v1)
```

As above, but calculate total within each level of `catvar`

```
egen newv3 = total(v1), by(catvar)
```

# Kategorisera variabler i kategorier

- Kommandot ”egen” innehåller flera användbara alternativ

*egen dia_year_cat=cut(dia_year), at(1996, 2005, 2013, 2020)  icodes label*

- Efter du delat upp datat i kategorier och vill dubbelkolla att det ser bra ut alt kolla hur det ser ut med andra variabler

*bysort dia_year_cat: sum dia_year*

*bysort dia_year_cat: sum age_at_dia*

# Ändra variabler eller kategorinamn

- Skapa exakt kopia av en variabel

*clonevar*

- Ändra numeriskt värde från ett till ett annat (missing går också bra)

*recode variable1 (0=1) (1=2) (2=.)*

# Spara och gå tillbaka till datat

- Spara den öppna filen, eller spara specifika variabler i den öppna filen

*save*

*savesome*

- Spara filen precis som den ser ut nu för att enkelt gå tillbaka till den senare

*snapshot save*

*snapshot restore*

*preserve/restore*

- Spara variabellistan/metadatat i excel (för att enkelt delas med kollegor)

*describe, replace clear*

*export excel using “Variables_in_data", firstrow(variables) replace*

# ÅÄÖ i datat

- Ibland finns variabler med labels eller strängar som innehåller ööå

*cd "/dataset folder/"*

*unicode encoding set iso-8859_10-1998*

*unicode translate "file_with_ööä.dta"*

# Strängvariabler

- Hitta specifika strängar i strängvariabler

*gen A_02 = regexm(ATC, "^A02")*

*gen A_03 = regexm(ATC, "^A03")*

Göra om strängen till en numerisk variabel

- *destring (fodelsear), generate (year)*

- Kapa strängvariabeln

*gen year=substr(dod_date_string,1,4)*

# Strängvariabler → numeriska med strängen som label

- Strängvariabler där man vill göra om variabeln till numerisk med behålla ”strängen” som en label till den nya numeriska variabeln (vanligt med data från kvalitetsregister där datat exporterats via INCA)

```
replace a_remiss_beskrivning = regexr(a_remiss_beskrivning, "NA", "")
/* Sätter NA som missing*/
encode a_remiss_beskrivning  , generate(referral_mode) label(a_remiss_beskrivning)
tab referral_mode, missing
codebook referral_mode
```

# Loopande av variabler, numeriska i ”ordning”

- Om det är numeriska data → forvalues, olika exempel nedan

```
forvalues i=1997/2018       {
                replace utb_cat=utb`i' if dia_year>`i' & utb`i'!="" & dia_year!=.
}

forvalues j=4(2)6 {
                use "lmed_`j'_wip.dta", clear
                /// other commands here..
                save "lmed_`j'_1rad1ind.dta", replace
}

forvalues i=1/30 {
                replace DIA`i'=strtrim(DIA`i')
                replace DIA`i'=ustrtrim(DIA`i’)
} /* Trimma diagnosvariablerna från patientregistret*/
```

# Loopande av variabler, i skapade kategorier/variabler, ej numeriska

- Om det finns en lista av kategorier → foreach
- Steg 1 → skapa listan som en lokal variabel

*local ATCgrupp A_02 A_03 A_04 A_06 A_07 A_10 B_01 B_03*

- Steg 2 → kör igenom listan

```
foreach ATCgrupp in A_02 A_03 A_04 A_06 A_07 A_10 B_01 B_03   {
                    tab ATC if `ATCgrupp'==1
}
```

# Loopande av variabler, kategorier

- Om man vill skapa variabler av olika kategorier som finns i datat, i detta fall olika ATC koder som finns i datat
- Steg 1 → skapa listan som en lokal variabel men använd levels

*levelsof ATC, local(levels)*

- Steg 2 → kör igenom din lista med foreach

```
foreach i of local levels {
        gen `i'=0 if ATC=="`i'"
}
```

# Skapa baslinjetabell

- Smidigt kommando för att skapa en baslinjetabell/tabell 1 tex för olika kön, fall/kontrollstatus etc

```
forvalues i=1/2 {
        baselinetable dia_year(cts) dia_year_cat age_num(cts) age_num_cat utb_cat_3cat if kon==`i',  by(kontroll) missing exportexcel(Baseline_`i', replace)
        baselinetable drug1 drug2(cts tab("\mean=mean (\sd sd)")) drug2(cts tab("p50(IQR p25-p75)")) drug2_cat if kon==`i', by(kontroll) missing exportexcel(Exp_`i', replace)
}
```

# Fler lunchseminarium

- **6/12 kl 12-13 Bootstrapping, vad är det och vad kan det vara bra för?**

*Kort om tanken bakom och hur det kan hjälpa oss skatta standard errors och konfidensintervall. När vi genom bootstrapping skapar nya stickprov från ett befintligt stickprov, känns det inte lite som att koka soppa på en spik?*

Henrik Holmberg / Per Liv, Region Västerbotten/ Institutionen för folkhälsa och klinisk medicin, Umeå universitet

- **18/1 kl 12-13 Nyfiken på interaktioner**

*Interaktioner är ett vanligt begrepp inom statistik och epidemiologi. Begreppet har dock olika betydelse beroende på sammanhang. I den här föreläsningen ger vi en introduktion till interaktioner och förklarar skillnaden mellan interaktioner inom statistiken och begreppen biological/causal-interactions inom epidemiologin. Vi berör begrepp som additiv och multiplikativ interaktion; effektmodifikation; Relative excess risk due to Interaction (RERI); och Attributable proportion due to interaction (AP) och förklarar dess användningsområden.*

Gabriel Granåsen, Region Västerbotten/ Institutionen för folkhälsa och klinisk medicin, Umeå universitet

Hör gärna av er om önskemål! Hemsida: https://www.canvas.umu.se/courses/2600

# FRÅGOR?

Tack!

Registercentrum Norr

Kliniska Studier Sverige Forum Norr

UMEÅ UNIVERSITY