Kontrol af koderreliabilitet med -datasignature- og -merge-

[2. udgave, september 2013]
Kim Mannemar Sønderskov
Institut for Statskundskab, Aarhus Universitet
ks@ps.au.dk

Efter indtastning af data kan man undersøge koderrealibiliteten – altså om indtastningen er foretaget uden (tilfældige) fejl. Ved små datamængder vil man typisk gennemgå hele det indtastede datasæt igen, men i større datasæt vil man oftest indtaste en udvalgt mængde data igen. Når dette er gjort kan man relativt let undersøge om de to datasæt er identiske og efterfølgende få et overblik over mængden af eventulle fejl samt lokalisere og rette fejlene. Ud fra mængden af fejl kan man vurdere koderrealibiliteten og om hele dataindtastningen bør kontrolleres.

I det følgende beskrives en procedure, der sammenligner et originalt datasæt med et kontroldatasæt, hvori (en del af) det originale datasæt er genindtastet. For at kontrollen kan lade sig gøre skal der være en id-variabel, der entydigt identificerer hver observation i begge datasæt. Det kan fx være løbenumre fra en spørgeskemaundersøgelse, cpr-numre i en registerundersøgelse, kommunenumre i en tværkommunalundersøgelse eller landenavne/iso-numere i en tværnational undersøgelse. Under alle omstændigheder skal hver observation have en unik værdi på denne variabel.

I nedenstående eksempler hedder variablen, der entydigt identificerer observationerne, id, selvom variablen naturligvis kan hedde hvad som helst (jf. afsnit 3.4.1 i Sønderskov, 2011). Der gennemgås to procedurer for kontrol af koderreliabilitet; én hurtig og nem procedure, der dog ikke altid kan anvendes og en lidt mere omfattende procedure, der kan anvendes i de fleste situaltioner.

Procedure 1: Sammenligning af datasæt med -datasignature-

Denne procedure kan anvendes i situationer, hvor man (midlertidigt) kan slette alle de observationer i originaldatasættet, der ikke er kontroltastet i kontroldatasættet således at kontroldatasættet og (det modificerede) originaldatasæt indeholder de samme observationer. Det kunne eksempelvis være, at man har kontroltastet de 100 observationer, der har de laveste id-værdier – fx værdierne 1-100. Det kunne også tænkes, at man havde kontroltastet alle observationer, hvorved de to datasæt også består af de samme observationer.

Hvis man ikke har kontroltastet alle observationer er første skridt i denne procedure at slette alle de observationer fra originaldatasættet, som ikke er kontroltastet i kontroldatasættet. Husk at gemme det originale datasæt inden du sletter observationer! Afsnit 3.7 i Sønderskov (2011) beskriver, hvordan udvalgte observationer slettes, men hvis vi eksempelvis vil slette alle observationer, som har højere værdier end 100 på **id** kunne det gøres således:

```
*sletter alle observationer, som har højere værdier end 100 på variablen id
drop if id>100
```

Nu kan originaldatasættet sammenlignes med kontroldatasættet. Dette kan gøres nemt med kommandoen datasignature. Kommandoen rapporterer en datasignatur i form af en talstreng. Talstrengen dannes på baggrund af en række karakteristika i datasættet, herunder antallet af observationer og variable, variablenes værdier, deres rækkefølge samt deres *storage type* (se afsnit 3.4.1 i Sønderskov, 2011 om storage type). Hvis de to datasæt er ens of dermed formentligt uden tilfældige fejl vil de have samme datasignatur. Kommandoen datasignature har en dialogboks, men kan nemt bestilles ved blot at taste datasignature:

```
*bestiller datasignatur for det aktive datasæt
datasignature
```

Dette gøres i begge datasæt og talstrengene kan herefter sammenliges. Hvis de ikke er fuldstændig identiske må man konstatere, at der er forskel mellem datasættene. Hvis man vil have et overblik over omfanget og eventuelt rette fejl kan man bruge nedenstående procedure.

Procedure 2: Sammenfletning af datasæt med -merge-

Denne procedure kan anvendes, hvis man ikke umiddelbart slette de observationer fra originaldatasættet, der ikke er kontroltastet (fx hvis man har kontroltastet et tilfældigt udvalg) eller i situationer, hvor man med datasignature har konstateret, at der er forskel mellem originaldatasættet og kontroldatasættet. Proceduren involverer følgende trin:

1) Omdøbe variablene i kontroldatasættet til andre navne end i originaldatasættet
2) Sammenflette de to datasæt
3) Undersøge om observationer har forskellige værdier på de samme variable og i givet fald identificere disse
4) Eventuel korrektion af fejlene.

Ad 1) Omdøbe variablene i kontroldatasættet til andre navne end i originaldatasættet

Årsagen til at variablene skal hedde have forskellige navne i de to datasæt er, at datasættene skal sammenflettes, hvorefter original og kontrolvariablene skal sammenlignes – dette nødvendiggør, at de har forskellige navne. Det skal dog bemærkes at id-variablen (som den eneste) ikke skal have forskellige navne i de to datasæt – den skal tværtimod hedde præcist det samme i de to datasæt. Med nedenstående to kommandoer omdøbes alle variable i datasættet, således at de alle får et 'X' foran variabelnavnet; herefter genomdøbes variablen id til det originale navn. Dette gøres således:

```
rename * X* //omdøber alle variable i datasættet ved at give dem prefixet X
rename X*id id* //genomdøber variablen id til den originale navn
```

Herefter skal kontroldatasættet lukkes og gemmes med et nyt navn (så du ikke overskriver det).

Ad 2) Sammenflette de to datasæt

Du kan nu sammenflette de to datasæt således, at variablene fra kontroldatasættet føjes til originaldatasættet. Sammenfletningen bevirker, at observationer med samme id-nummer flettes sammen. Dette er illustreret nedenfor i Figur 1 med et simpelt originaldatasæt (vist til venstre). Den fælles id-variabel

hedder id. Derudover består originaldatasættet af én variabel (var1) og fem observationer. I kontroldatasættet (vist i midten) hedder var1 Xvar1, og det indeholder kun tre observationer (observationerne med id-numrene 1, 3 og 5). Det fremgår i datamatricen til højre, at sammenfletningen har sørget for at placere observationerne fra kontroldatasættet på de rigtige pladser i det sammenflettede datasæt.

Figur 1: Sammenfletning af to datasæt til ét

	id	var1
1	1	200
2	2	100
3	3	150
4	4	75
5	5	175

	id	Xvar1
1	1	200
2	3	150
3	5	175

	id	var1	Xvar1
1	1	200	200
2	2	100	.
3	3	150	150
4	4	75	.
5	5	175	175

En sådan sammenfletning kan foretages med kommandoen merge. merge kan bruges til mange slags sammenfletningerne (eksempelvis sammenfletning af surveydata fra en række lande med landedata fra de samme lande), men her vil vi bare bruge den til at sammenflette to datasæt.

Først skridt er at åbne originaldatasættet. Sammenligningen af variablene, der beskrives nedenfor bliver nemmere, hvis originaldatasættet er ordnet således at id-variablen står først (længst til venstre) i datasættet. Hvis id-variablen ikke allerede står først kan dette gøres med kommandoen order (jf. Sønderskov 2011: afsnit 4.1.3):

```
*placerer variablen id først i datasættet
order id
```

Næste skridt er selve merge-kommandoen. Den har tilknyttet en hjælpsom dialogboks (db merge eller Data > Combine datasets > Merge two datasets). Under Key variables: (match variables) indsættes id-variablen og under Filename of dataset on disk: skal man finde kontroldatasættet.

Denne form for merge-kommando forudsætter (som nævnt ovenfor), at hver observation har en unik værdi på id-variablen – hvis dette ikke er tilfældet vil Stata meddele følgende efter eksekveringen af kommandoen: "variable id does not uniquely identify observations in the master data"; master kan være erstattet af using, hvis det er kontroldatasættet, der indeholder ikke-unikke værdier på id-variablen. Boks 1 beskriver, hvordan man lokaliserer ikke-unikke id-værdier.

Hvis hver observation har unikke id-værdier i begge datasæt vil eksekveringen af kommandoen sammenflette de to datasæt og rapportere resultatet af sammenfletningen. Indholdet af rapporten giver et første fingerpeg om der er problemer med koderreliailiteten. Rapporten fortæller, hvor mange observationer der er matchede – altså hvor mange observationer, der har de samme værdier på id-variablen i de to datasæt – og eventuelt, hvor mange der ikke er matchede. Antallet af matchede variable står ud for matched og dette antal skulle gerne svare til antallet af observationer i kontroldatasættet. Samtidigt skulle antallet ud for not matched – from master gerne svare til antallet af observationer, der

ikke er kontroltastet. Hvis ovenstående ikke er tilfældet skyldes det, at der findes id-værdier i kontroldatasættet, som ikke findes i originaldatasættet – dette antal er rapporteret ud for not matched – from using. Det vil som regel være nødvendigt at lokalisere disse og korrigere fejlene. Boks 2 beskriver hvordan sådanne observationer lokaliseres. Hvis rapporten viser det forventede antal sammenflettede observationer, kan man gå videre og undersøge koderreliabiliteten.

Ad 3) Undersøge om nogle observationer har forskellige værdier og i givet fald identificere disse

Næste trin er at sammenligne værdierne mellem variablene fra originaldatasættet (fx var1) og kontroldatasættet (fx Xvar1). Den nemmeste og mest fejlsikre måde at gøre dette på at ved hjælp af et kommandoloop og foreach-kommandoen (jf. afsnit 7.7 i Sønderskov, 2011). Vi skal dog bruge lidt anden udformning af foreach end beskrevet i Sønderskov – vi skal bruge en foreach-kommando, der forstår implicitte henvisninger til variabelnavne.

Følgende start på foreach-kommandoen forstår implicitte henvisninger til variabelnavne:

```
foreach var of varlist var1-var3 {
```

Dette vil starte et loop, hvor makroen var først antager variabelnavnet var1 og derefter alle variabelnavne fra var1 og til og med var3 (jf. afsnit 3.6.1 i Sønderskov, 2011). Hvis eksempelvis datasættet er arrangeret således, at var2 står umiddelbart efter var1 og før var3 vil loopet altså køre tre gange.

Dette udnyttes i nedenstående loop, hvor var løbende antager hvert af variabelnavnene fra originaldatasættet (med undtagelse af id, der står først i datasættet). I anden linje bedes om at liste id-nummeret og værdierne på variablene for alle de observationer, der ikke har identiske værdier på hvert par af variable fra de to datasæt (med undtagelse af observationer, der ikke er kontroltastet – derfor betingelsen om at _merge skal være 3 – jf. Boks 2).

```
foreach var of varlist var_først-var_sidst {
list id `var' X`var' if `var' != X`var' & _merge ==3
}
*var_først skal være første variabel fra originaldatasættet – efter id-variablen
*var_sidst skal være sidste variabel fra originaldatasættet
```

Ad 4) Eventuel korrektion af fejlene

Med udgangspunkt i outputtet fra list-kommandoen kan man korrigere eventuelle fejl i variablene fra orginaldatasættet.

Boks 1: Lokalisering af observationer med ikke-unikke værdier på en variabel

Hvis man vil finde værdier, der optræder mere end en gang på en variabel kan det gøres ved først at sortere den pågældende variabel, sådan (jf. afsnit 2.2.5 i Sønderskov, 2011):

```
*sorterer variablen id stigende:
sort id
```

Herefter beder man Stata om at liste alle observationer, der har samme værdi som den næste observation. Dette gøres med nedenstående kommando, hvor det udnyttes, at man kan henvise til en værdi på en variabel i kantede parenteser. id[1] henviser eksempelvis til værdien på id for observation nummer 1. Samtidigt udnyttes det at _n betyder "nuværende observationsnummer". Når Stata gennemsøger datasættet (observation for observation) efter observationer, der opfylder nedenstående if-sætning, vil if-sætningen, når Stata undersøger første observation lyde if id[1] == id[2] osv.

```
*oplister observationer, der har samme id-værdi som den næste observation
list id if id[_n] == id[_n+1]
```

list-kommandoen vil opliste samtlige observationer, der har samme id-værdi som den næste observation, og man kan herefter korrigere indholdet af variablen.

Boks 2: Identificere observationer der ikke blev sammenflettet

De er nemt at finde observationer, der ikke blev sammenflettet med merge-kommandoen. Som standard genererer merge en variabel med navnet _merge efter eksekvering. Afhængigt af, hvorvidt observationen blev sammenflettet eller ej og årsagen til at observationen eventuelt ikke blev sammenflettet, får observationerne forskellige værdier på denne variabel. Observationer, der er sammenflettet får værdien 3, mens observationer, der ikke er sammenflettet fordi de kun eksisterede i master-datasættet (eller original datasættet i nærværende eksempel) får værdien 1. Observationer der kun eksisterede i using-datasættet (kontroldatasættet) får værdien 2. I nærværende eksempel er vi interesseret i at finde observationer, der kun eksisterede i kontroldatasættet – altså observationer der har værdien 2 på variablen _merge.
Disse kan listes således:

```
*oplister observationer med værdien 2 på _merge
list id if _merge==2
```

Referencer

Sønderskov, K.M. (2011). Stata – En praktisk introduktion. København: Hans Reitzels.