Note om Monte Carlo eksperimenter

Mette Ejrnæs og Hans Christian Kongsted
Økonomisk Institut, Københavns Universitet

19. september 2003

Denne note er skrevet til kurset Økonometri 1 på 2. årsprøve af polit-studiet. Formålet med noten er at forklare ideen med at lave simulationseksperimenter, også kaldet Monte Carlo eksperimenter. Simulationseksperimenter er velegnede til at illustrere centrale statistiske og økonometriske begreber. Metoden har desuden vist sig utroligt nyttig indenfor stort set alle grene af økonometrien til at efterprøve egenskaberne for estimatorer og test. I noten lægges specielt vægt på, hvordan et Monte Carlo eksperiment laves i praksis. Derfor er der medtaget et gennemgående eksempel, som detaljeret beskriver, hvordan et eksperiment kan konstrueres og udføres i SAS.

1. Hvad er et Monte Carlo eksperiment?

Et Monte Carlo eksperiment er baseret på simulationer af en konkret statistisk model. Ved at simulere et udfald fra modellen bestående af et bestemt antal (fx $n = 100$) observationer opnår man et "kunstigt" datasæt. På grundlag af datasættet kan man så beregne de størrelser, der har interesse for eksperimentet. Det kunne fx være OLS-estimatet af en koefficient i en lineær regressionsmodel. Simulerer man et (stort) antal udfald (fx $M = 10.000$) fås en fordeling af parameterestimater set over de kunstige datasæt ("replikationerne"). Man kan fx beskrive fordelingen ved at tegne et histogram og beregne dens gennemsnit og varians.

Monte Carlo eksperimentet er helt parallelt til det "tankeeksperiment", der ligger til grund for det meste statistik og økonometri, jf afsnit 2.5 og 3.3 i Wooldridge (2003). Man forestiller sig, at det konkrete sæt af faktisk observerede data, man er i færd med at analysere, blot er ét blandt mange hypotetiske udfald. Under ganske bestemte forudsætninger kan vi så udlede resultater fx for OLS-

estimatorens egenskaber set over disse hypotetiske udfald og bruge disse egenskaber til at vurdere estimaterne med. Et (vigtigt) eksempel: Under Gauss-Markov antagelserne vil OLS være den bedste lineære og middelrette estimator, hvilket betyder at set over et antal hypotetiske udfald vil OLS-estimaterne i gennemsnit ramme den sande værdi af parametrene. De vil tilmed gøre det med den mindste varians af alle tænkelige lineære og middelrette estimatorer.

Ideen med Monte Carlo eksperimentet er at erstatte de hypotetiske udfald med konkrete men kunstige replikationer. Det kræver at man er villig til at specificere den statistiske model der genererer de kunstige datasæt, helt ned til at vælge konkrete værdier af modellens parametre (fx $\beta_0 = 7$ og $\beta_1 = 0.25$ i en simpel lineær regressionsmodel) og bestemte fordelinger af de stokastiske variabler i modellen (fx at fejlleddet u_i fremkommer som uafhængige og identiske trækninger fra en standard normalfordeling). Monte Carlo eksperimentets resultat vil i almindelighed afhænge af de parameterværdier og fordelinger man vælger. I nogle tilfælde kan man dog opnå resultater, der er invariante i forhold til visse aspekter af den valgte model.[1]

For at illustrere denne ide vil vi se på et gennemgående eksempel. Antag at man ønsker at bestemme den forventede startløn for en nyuddannet økonom. Man ringer derfor rundt til $n = 100$ tilfældigt udvalgte nyuddannede økonomer (som er i beskæftigelse) og spørger til deres startløn. Derefter beregnes den gennemsnitlige startløn i stikprøven som et estimat for middelværdien i fordelingen af startlønninger. Vi er interesserede i at vide, om gennemsnittet er en god eller dårlig estimator i en given model. Eller formuleret på en anden måde: Hvis man tilfældigvis havde kontaktet $n = 100$ andre nyuddannede økonomer og herved fået et andet gennemsnit, hvor forskelligt må man forvente at de to estimater vil være og hvor tæt kan man antage, at gennemsnitslønnen i den faktisk indhentede stikprøve ligger på den sande (men ukendte) middelværdi i fordelingen af startlønninger. For at besvare dette er man interesseret i at kende fordelingen af estimatoren (eller i hvert fald dens middelværdi og varians), set over alle de stikprøver, man potentielt kunne have fået.

Normalt vil det være krævende og omkostningsfyldt at begynde at indsamle nye data. I praksis vil man som regel afholde sig fra det. Fordelen ved at lave simulationseksperimenter er, at det er let og (stort set) gratis at få fat i nye kunstige datasæt. Man skal dog være villig til at levere en fuld

[1]Fx vil resultaterne ofte være uafhængige af fejlleddets varians, som i givet fald blot kan normaliseres til 1.

specifikation af den model, der genererer startlønninger, den såkaldte datagenererende proces (DGP).

I eksemplet vil vi arbejde med en antagelse om, at startlønningerne stammer fra uafhængige og identisk fordelte trækninger fra en normalfordeling. På DJØFs hjemmeside www.djoef.dk oplyses det, at den "vejledende startløn" for en privatansat, nyuddannet økonom pr. 1. januar 2003 er kr. 28.500 om måneden, hvilket vi vil antage er den "sande" middelværdi i lønfordelingen. Vi antager desuden at den sande lønfordeling har en standardafvigelse på kr. 1.500. Hermed er lønfordelingen fuldt specificeret.

Opgave: Hvad er - under disse antagelser - sandsynligheden for, at en tilfældigt udvalgt nyuddannet økonom har en startløn på mere end kr. 31.500?

Man kan også forestille sig, at man overvejer at estimere den forventede startløn på en alternativ måde (f.eks. som medianlønnen eller som gennemsnittet af den største og mindste startløn i stikprøven).[2] Man er så interesseret i at vide, hvilken af de foreslåede estimatorer, der er den mest præcise (fx hvilken der har den mindste varians). Ved at simulere estimatorerne kan man sammenligne deres fordelinger i Monte Carlo eksperimentet og herved se, hvilken som har den mindste varians.

Et andet formål med at lave simulationer kan være at undersøge, hvor mange observationer – hvor stort n - man skal bruge for at få et "rimeligt" præcist estimat.

Endelig kan man være interesseret i at undersøge, hvad der sker med fordelingen af estimaterne, hvis f.eks. Gauss-Markov antagelserne ikke er opfyldt.

I nogle tilfælde kan man nå frem til analytiske svar på disse spørgsmål. Det vil fx være muligt i den relativt simple problemstilling, vi har skitseret her. Fordelingen af et stikprøvegennemsnit er gennemgået i afsnit 6.2 af "Teoretisk statistik for økonomer" (sætning 6.1 og 6.2) og vi vil sammenligne vores simulationsresultater med de analytiske resultater senere i noten. Men ofte er man interesseret i at få en ide om, hvordan fordelingen af estimaterne ser ud i mere komplicerede sammenhænge, hvor de simple forudsætninger ikke holder og det kan derfor være vanskeligt at

[2] Overvej under hvilke antagelser om fordelingen af startlønninger, at medianen vil være en rimelig estimator i dette eksempel.

opnå analytiske resultater. Simulationseksperimenter vil ofte være et værdifuldt værktøj til at belyse dette.

Igen er det vigtigt at understrege, at man kun kan lave simulationer, hvis man er villig til at specificere den model, der genererer de kunstige data (DGP'en) og at resultaterne almindeligvis vil afhænge heraf. Man må derfor godtgøre, at den valgte DGP er relevant for den problemstilling, man ønsker at belyse.

Opgave: Er antagelserne om lønfordelingen i eksemplet realistiske? Hvilke andre fordelinger kunne du foreslå som model for lønfordelingen?

2. Hvordan laves et Monte Carlo eksperiment i praksis?

I dette afsnit gennemgås, hvordan et simulationseksperiment laves. Antag, at vi ønsker at simulere fordelingen af en bestemt estimator baseret på n observationer, givet ved $\{y_1, y_2, ..., y_n\}$. Vi må først specificere modellen, der genererer disse data. Den model vi arbejder med skal være fuldt parametriseret. Det betyder som nævnt, at man både skal antage hvilken fordeling observationerne stammer fra (f.eks. normalfordelingen) og den eksakte værdi af parametrene (f.eks. middelværdi og varians).

I løneksemplet vil vi som nævnt antage, at startlønningerne er uafhængige og normalfordelte omkring en middelværdi og med variansen σ^2. Vi kan så skrive

(1) $$y_i = \mu + \sigma\varepsilon_i \qquad i = 1, ..., n,$$

hvor ε_i er standardiseret uafhængige normalfordelt variable $(iidN(0,1))$. Vi antager her, at vi kender de "sande parametre" μ og σ^2.

Trin 1: Konstruér de "kunstige" data

I dette trin konstrueres et datasæt fx for $n = 100$ personer. I tilfældet med startlønningerne vil vi antage, at den månedlige startløn y_i er givet (i 1.000 kr.) ved model (1) og at parameterværdierne er

$$\mu = 28,5, \quad \sigma=1,5$$

Vi kan nu generere fiktive startlønninger ved at trække 100 tilfældige tal fra en standardiseret normalfordeling: $e_1, e_2, ..., e_{100}$. I praksis bruger man pseudo-tilfældige trækninger fra en computerbaseret generator af tilfældige tal[3] og startlønnen for den fiktive person nummer *i* konstrueres som

$$y_i = 28,5 + 1,5e_i .$$

SAS koden:

Simulationseksperimenter kan laves i de fleste statistikprogrammer. Det afgørende er, at programmet kan lave en løkke og generere "tilfældige" tal.[4] I dette kursus benyttes SAS og simulationen udføres ved hjælp af proceduren Proc IML.[5] Inden for denne procedure er man i stand til at lave matrixregning samt at lave løkke omkring et sæt af beregninger.

I det første trin skal data dannes. Før data dannes er det en fordel at lave en variabel, som angiver, hvor mange observationer, n, vi vil lave i hvert af de kunstige datasæt (kaldet antalobs i SAS-koden). De "tilfældige" tal genereres her med SAS-funktionen NORMAL. Når man laver tilfældige tal er det ofte en fordel at kunne rekonstruere præcis samme udfald ved en senere lejlighed. Det opnår man ved at give SAS-funktionen et såkaldt "seed" (et "frø"). I praksis danner man først en vektor af et-taller med samme dimension som den ønskede vektor af tilfældige tal. I eksemplet er det en vektor med n elementer. Dernæst vælger man en "seed" i form af et heltal og ganger tallet på hele vektoren.[6] I nedenstående programstump er 117 valgt som seed.

[3] At tallene er pseudo-tilfældige betyder, at de i virkeligheden produceres af en deterministisk computeralgoritme, men på en måde så de approximerer tilfældige trækninger "godt nok" til vores (og de fleste andre) formål.

[4] SAS giver mulighed for at trække pseudo-tilfældige tal fra en række forskellige fordelinger, fx normal-, t-, F- og uniformt fordelte trækninger.

[5] En kort beskrivelse af Proc IML findes i en note på øvelseshjemmesiden. Mere udførlig hjælp findes i SAS Help under "Help on SAS Software products".

[6] Reelt har kun det første element i vektoren nogen betydning. Det fastlægger hele sekvensen af "tilfældige" tal inden for et givet kald af IML. Også når vi senere kører programsekvensen *flere gange* indenfor *samme IML kald* har seed'en kun betydning første gang. Ønsker man ingen seed kan man angive 0. SAS vælger så selv hvor sekvensen af tilfældige tal skal starte, hvilket varierer hver gang programmet køres.

SAS koden til at konstruere ét fiktivt datasæt på 100 observationer kan se således ud:

```
Proc IML;
  antalobs =  100;    * antal observationer i datasættet;
   mu = j(antalobs,1,28.5) ; * middelværdivektor ;

   seedvct = j(antalobs,1,1) ;  * Samme dimension som vektor af tilfældige tal ;
   seedvct = 117*seedvct ;      * Seedværdien er sat til 1.

   * Laver en vektor af uafhængige standard normal fordelte variabler ;
   e   = normal(seedvct) ; * Dimension af e bestemmes af seedvct ;

   * vektor af kunstige løndata fra den ønskede fordeling ;
   y   = mu + 1.5 * e ;

quit;
```

Trin 2: Find estimaterne

I dette trin udregnes de relevante estimater. I eksemplet med startlønninger var den parameter, vi er interesseret i, middelværdien μ. I dette eksempel vil vi sammenligne to estimatorer nemlig gennemsnittet m_1 og gennemsnittet af den største og mindste observation m_2:

$$m_1 = \frac{1}{100}\sum_{i=1}^{100} y_i$$

$$m_2 = \frac{1}{2}\left(\min_{i=1,,,100}(y_i) + \max_{i=1,,,100}(y_i)\right)$$

Når estimaterne er udregnet gemmes de for hvert genereret datasæt. Estimaterne indlæses direkte i matricer, der rummer resultater for alle M replikationer. Dette er smart når man går til trin 3.

SAS-koden til udregning af estimaterne:

```
  m1[j,1]=sum(y)/antalobs;     * estimatet m1 (gennemsnittet);
  m2[j,1]=1/2*(min(y)+max(y)); * estimatet m2 (gns. min og max);
```

Trin 3: Gentag trin 1 og 2

Ideen med trin 3 er, at man nu har mulighed for at trække et nyt datasæt og estimere parametrene på baggrund heraf. Dette gøres ved at gentage trin 1 og 2. Det ønskede antal replikationer afhænger af

det specifikke eksperiment. Generelt vil man gerne have så mange replikationer som muligt, da det øger præcisionen i bestemmelsen af fordelingen af estimatorerne. Omkostningen ved et meget højt antal replikationer er, at det kan tage lang tid at udføre eksperimentet (dette afhænger også af computeren). I mange tilfælde vil 10.000 være et rimeligt antal.

SAS-kode:
I dette trin skal man lave en løkke, som i hvert trin genererer et datasæt og estimerer parametrene som i trin 1 og 2, og gentager dette 10.000 gange. Løkken startes ved at skrive: ”do j=1 to antalrep;” og afsluttes ved at skrive: ”end;”. Løkken løber nu over indekset j, og de kommandoer, som står mellem ”do..” og ”end;” udføres nu ”antalrep” gange. I begyndelsen af programmet er ”antalrep” sat lig med 10.000.

Derefter skal man sørge for, at man for hver simulation får gemt sine estimater. Dette kan gøres ved, at man i starten af sit program laver matricer til estimaterne. Dimensionen af disse skal svare til antallet af simulationer. I dette tilfælde laves to vektorer m1 og m2, som hver har dimensionen 10.000×1. Dernæst laver man en løkke på samme måde som i trin 1.[7]

SAS-kode til løkke over antal replikationer:

```
antalrep = 10000;   * antal replikationer i simulationen;
*initialiser vektorer;
m1 = j(antalrep,1,.);  * vektorer til at gemme estimaterne i;
m2 = j(antalrep,1,.);
do j=1 to antalrep ;  * løkke over simulationer;
       .
       .
       .
end;
```

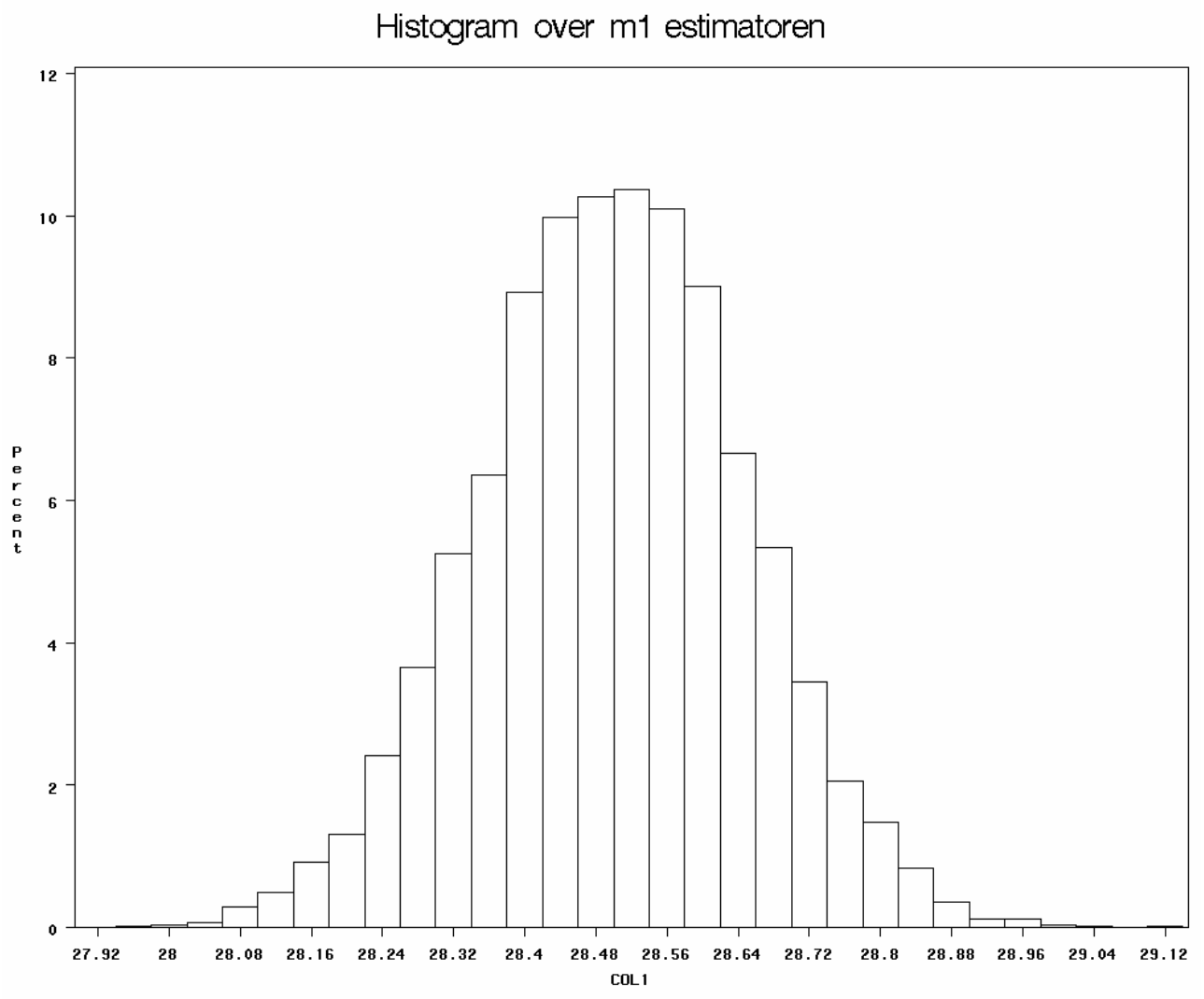
[7] Hvis man arbejder med meget høje antal replikationer og/eller har brug for at gemme mange estimater for hver replikation kan det af hensyn til computerens kapacitet være nødvendigt at beregne fx gennemsnit og varians af estimaterne løbende i stedet for at gemme hele sekvensen i en matrix.

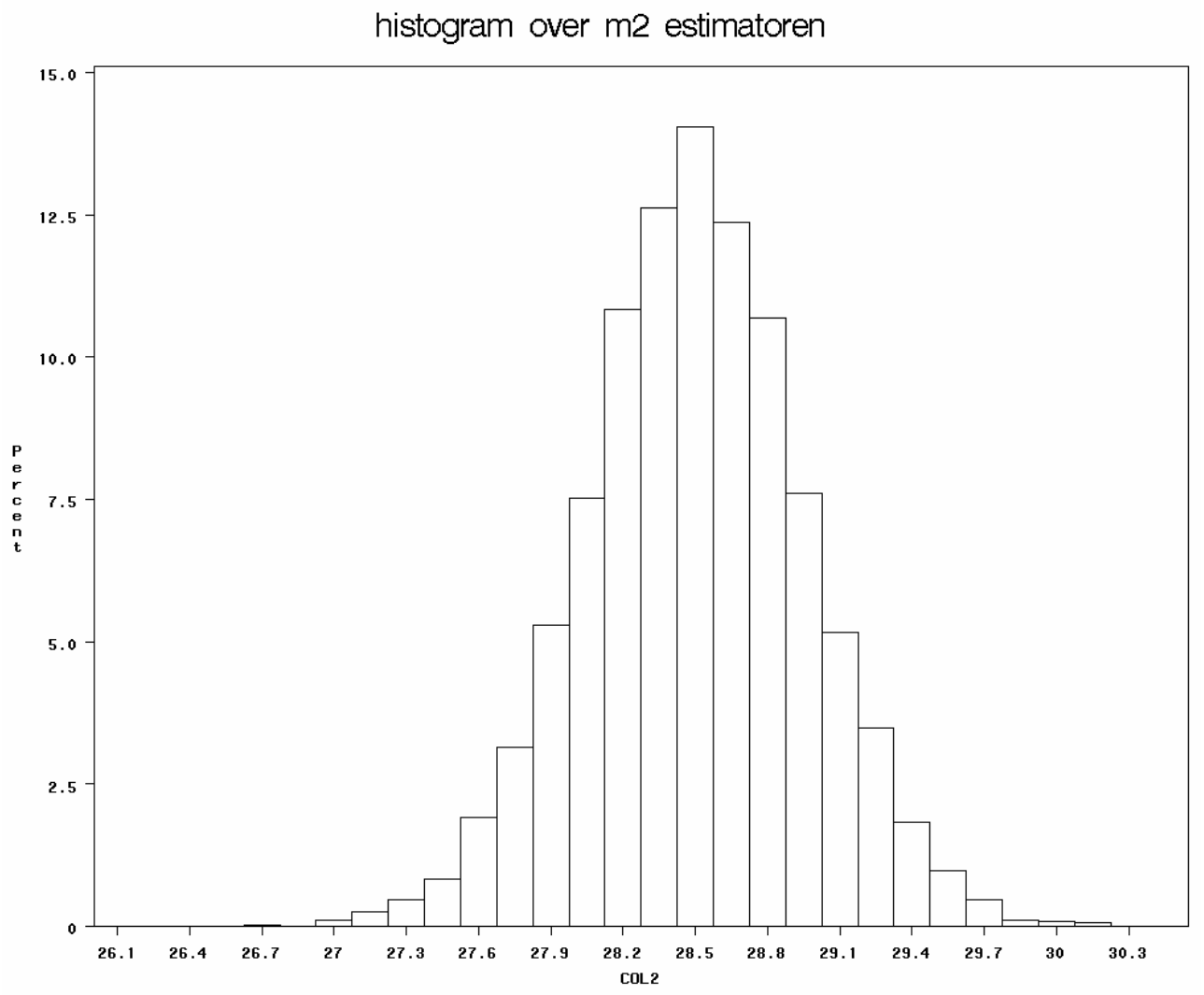
Trin 4: Analyse af estimater baseret på simulerede data

Efter at have udført trin 1-3 vil man have et antal (f.eks. 10.000) "realisationer" af estimaterne. Ud fra disse kan man så undersøge fordelingen af estimaterne eller udregne middelværdi og varians på estimatet.

Figurne nedenfor viser histogrammer af de to sæt af estimater. Begge estimater er centerede omkring den sande værdi 28,5. Det ses også, at fordelingen af estimaterne ligner en normalfordeling. Den beregnede middelværdi og varians for de to fordelinger er angivet i tabellen.

Figur: *Histogram over estimaterne baseret på n=100 observationer og M=10.000 simulationer*





For at undersøge hvad der sker med estimaterne, hvis vi har et datasæt med færre observationer, gentages simulationseksperimentet, hvor vi kun har hhv. 50 individer og 10 individer i hvert datasæt. Det sker i praksis ved at antalobs ændres til 50 og dernæst til 10. I tabellen nedenfor er middelværdi og varians for fordelingerne af de to estimater angivet.

Tabel: *Middelværdi og varians af de to estimatorer baseret på M=10.000 simulationer*

	m_1	m_2
N=100		
Middelværdi	28,4990	28,5015
Varians	0,0223	0,2089
N=50		
Middelværdi	28,4993	28,4988
Varians	0,0443	0,2445
N=10		
Middelværdi	28,4975	28,4894
Varians	0,2209	0,4116

Sammenligner man middelværdien af estimaterne, kan man se, at for begge estimater (og for alle værdier af n) er middelværdien tæt på den "sande" middelværdi, som er 28,5. Dette stemmer overens med, at begge estimatorer er middelrette, dvs. uanset n vil estimatorerne have middelværdi lig den sande parameterværdi. Vil man yderligere forbedre simulationseksperimentet, fx bestemmelsen af middelværdien, skal man sætte antallet af replikationer op.

Hvis man sammenligner variansen af de to estimatorer i tabellen, kan man se, at variansen er størst for m_2.

Opgave: Kunne vi på forhånd have sagt noget om, hvilken af de to estimatorer, der har den mindste varians?[Hint: Kan vi bruge Gauss-Markov teoremet til sammenligningen?]

Det fremgår også af tabellen, at variansen af estimaterne stiger, når n falder.

Opgave: For estimatoren m_1 kan den teoretiske varians udregnes ud fra afsnit 6.2 i "Teoretisk statistik for økonomer". Beregn den teoretiske varians for hver værdi af n og sammenlign med simulationsresultaterne.

SAS-kode:
For at analysere estimaterne er det lettest at lave matricerne med estimater om til et SAS datasæt. Dette kan gøres ved at bruge kommandoen ”create” som laver et nyt datasæt.

```
dd=m1||m2;                    * datamatricen med de to estimater;
create hist from dd ;         * udskriver matricen til et datasæt;
append from dd;
```

Dernæst kan estimaterne analyseres ved at benytte Proc Univariate.

3. Afrunding

Vi har skitseret ideen med at anvende simulationseksperimenter i Økonometri 1. De opfylder en række formål, fx at efterprøve teoretiske resultater, hvor disse kan udledes analytisk, og helt erstatte analytiske resultater, hvor disse er vanskelige eller umulige at opnå. Vi vil se eksempler på begge typer af analyser i forelæsningerne og i øvelsesopgaverne.

Noten giver også en skabelon i SAS til at lave egne simulationseksperimenter. SAS-stumperne i noten er samlet til et program, der kan ses i appendix og ligger som fil på hjemmesiden. Bemærk at programmet primært er skrevet med et pædagogisk formål og at det ikke nødvendigvis er særligt kompakt eller efficient i sin opbygning.

4. Supplerende litteratur

Vil man læse mere om simulationseksperimenter vil et godt sted at starte være: J. Johnston og J. DiNardo: ”Econometric Methods”, 4. udgave, afsnit 11.1. En mere avanceret kilde er D.F. Hendry: ”Monte Carlo experimentation in econometrics”, i Z. Griliches og M. Intriligator: Handbook of econometrics, kapitel 16.

Appendix:

SAS-program

Det samlede program til at lave simulationseksperimentet beskrevet i noten.

```
dm "clear out";
dm "clear log";

Proc IML;
  antalobs =  100 ;          * antal observationer i datasættet;
  antalrep = 10000;          * antal replikationer i simulationen;

  *initialiser vektorer;
  m1 = j(antalrep,1,.);     * vektorer til at gemme estimaterne i;
  m2 = j(antalrep,1,.);

  mu = j(antalobs,1,28.5) ; * middelværdivektor ;

  seedvct = j(antalobs,1,1) ;  * Samme dimension som vektor af tilfældige tal ;
  seedvct = 117*seedvct ;      * Seedværdien er sat til 117
                                 Kun 1. element i vektoren får betydning ;

  do j=1 to antalrep ;       * løkke over replikationer ;

    * Genererer trækning af et datasæt på antalobs observationer ;

    * Laver en vektor af uafhængige standard normal fordelte variabler ;
    e   = normal(seedvct) ; * Dimension af e bestemmes af seedvct ;
                            * Kun 1. element i seedvct har betydning ;
    * vektor af kunstige løndata fra den ønskede fordeling ;
    y   = mu + 1.5 * e ;

          * estimaterne gemmes i de to bogføringsvektorer ;
    m1[j,1]=sum(y)/antalobs ;       * estimatet m1 (gennemsnittet);
    m2[j,1]=1/2*(min(y)+max(y));    * estimatet m2 (gennemsnit af min og max);

  end;                        * løkken over replikationer slutter her ;

  dd=m1||m2;                  * datamatricen med de to estimater;
  create hist from dd ;       * udskriver matricen til et datasæt;
  append from dd;
run;
quit; * Her forlades IML modulet.
De beregnede estimater m1 og m2 ligger i datasættet hist ;

proc univariate data=hist;    * deskriptiv statistik på m1;
var col1;
title "Histogram over m1 estimatoren";
histogram ;
run;

proc univariate data=hist;    * deskriptiv statistik på m2;
var col2;
title "histogram over m2 estimatoren";
histogram ;
run;
```