

## Datorövning 2

### Statistikens Grunder 2

#### Syfte

1. Lära sig beräkna konfidensintervall och täckningsgrad
2. Lära sig rita en exponentialfördelning
3. Lära sig illustrera centrala gränsvärdessatsen

#### Exempel

##### Beräkna konfidensintervall och täckningsgrad

Vi börjar med att skapa 100 stickprov med 100 observationer i varje stickprov. Dessa observationer är slumptal som kommer från en likformig fördelning mellan 0 och 1. Vi gör det med koden:

```
data work.random;  
do sample=1 to 100;  
do obs=1 to 100;  
x=ranuni(801120);  
if 0<= x <0.1 then nummer=1;  
if 0.1<= x <=1 then nummer=0;  
output;  
end;  
end;  
run;
```

Vi kallar data-setet "*work.random*". Vi vill ha 100 stickprov ("*sample*") och 100 observationer ("*obs*"). Variabeln med slumptal kallar vi "*x*". Talet inom parantesen anger fröet (seed). Det behövs alltid ett frö för att generera slumptal, så att algoritmen har en startpunkt. Jag har använt mitt födelsenummer som frö och jag vill att ni ska använda ert födelsenummer som frö. Eftersom samma frö ger exakt samma slumptal, är det intressantare om alla får olika slumptal. Därefter skapar vi en variabel som heter "*nummer*" och som får värdet 1 om slumptalet ligger mellan 0 och 0.1, och värdet 0 för övrigt. Detta gör att variablen "*nummer*" kan ses som en observation från en population där  $\pi = 0.1$ . Eftersom vi har två "*do*-" kommandon måste vi ha två "*end*".

Nu vill vi beräkna medelvärdet av variabeln "*nummer*" i varje stickprov och dessutom göra ett konfidensintervall. Vi skriver

```
proc means data=work.random CLM mean alpha=0.05;  
output out=work.intervals LCLM=lower UCLM=upper;  
var nummer;  
by sample;  
run
```

Vi anger "*CLM*", "*mean*" och "*alpha=0.05*" för att SAS ska räkna ut ett 95%-igt konfidensintervall och ett medelvärde för variabeln "*nummer*". Vi vill att SAS ska göra detta för varje stickprov, så vi skriver "*by=sample*". Vi specificerar att vi vill spara nedre klassgränsen som "*lower*" och övre klassgränsen som "*upper*" i data-setet "*work.intervals*".

Vi kommer nu att få 100 konfidensintervall. Vi vet att populationsproportionen är 0.1 och vill därför att konfidensintervallen täcker 0.1. Vi undersöker hur många av intervallen som täcker 0.1 med koden

```
data work.intervals;
set work.intervals;
if lower<= 0.1 and upper =>0.1 then coverage=1;
else coverage=0;
run;
```

Genom att använda "*data....set...*" kan man modifiera variabler som man har i sitt data-set. Här bildar vi en ny variabel, som heter "*coverage*" (täckning), genom att använda oss av de befintliga variablerna "*lower*" och "*upper*". Vi vill att variabeln "*coverage*" ska anta värdet 1 om intervallet täcker 0.1 och 0 annars.

Nu vill vi räkna hur många av intervallen som täckte 0.1. Vi skriver

```
proc freq data=work.intervals;
tables coverage;
run;
```

Hur många intervall förväntas täcka 0.1?

### Rita en exponentialfördelning

Vi vill nu rita en täthetsfunktion för en exponentialfördelning med parameter  $\lambda = 1$ . Vi gör på samma sätt som vi gjorde på Datorövning 2, Statistikens Grunder 1, när vi presenterade binomial- och normalfördelningen grafiskt.

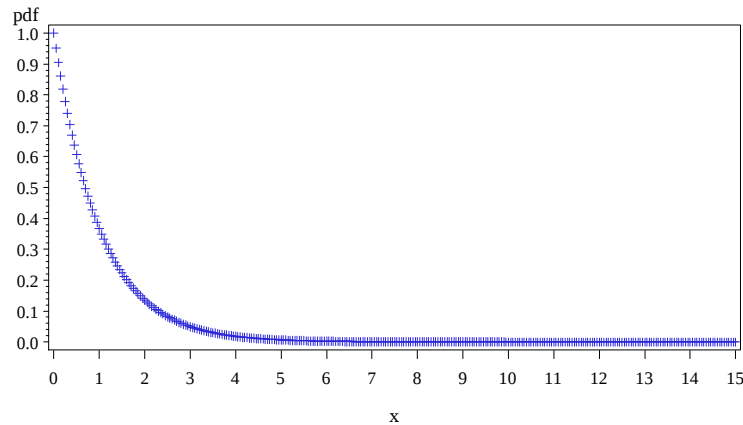
```
data work.exp;
do x = 0 to 15 by 0.05;
pdf = pdf('expo',x,1);
output work.exp;
end;
proc print;
run;
```

Med koden ovan skapar vi en variabel som går från 0 till 15 med stegen 0.05. Därefter talar vi om att vi vill beräkna värdet på täthetsfunktionen för varje x-värde. Efter "*output*" anger vi att vi vill spara dessa funktionsvärden i data-setet "*work.exp*".

För att illustrera fördelningen använder vi koden

```
proc gplot data=work.exp;
plot pdf*x;
run;
```

Vilket ger grafen



som visar hur en exponentialfördelning med parameter  $\lambda = 1$  ser ut.

### Illustrera centrala gränsvärdessatsen

Vi vill nu skapa slumpstal som kommer från en exponentialfördelning med parameter  $\lambda = 1$ . Med koden nedan skapar vi 1000 stickprov med 10 observationer i varje stickprov.

```
data work.randomexp;  
do sample = 1 to 1000;  
do obs = 1 to 10;  
x = ranexp(801120)/1;  
output;  
end;  
end;  
run;
```

Variabeln "x" ska innehålla alla våra värden. Vi skriver "*ranexp*" för att tala om att vi vill ha slumpstal, "*random numbers*", från exponentialfördelningen. Talet inom parantesen anger återigen fröet (seed). Kommandot "/1" talar om att parameterns värde är 1. Vi skriver "*output*" för att vi vill spara datat. Eftersom vi har två "do-" kommandon måste vi ha två "end".

Vi ska nu beräkna medelvärde för varje stickprov. Vi kommer alltså få 1000 medelvärden. Vi skriver koden:

```
proc means data=work.randomexp mean ;  
output out=work.exponential_means mean=xbar ;  
var x;  
by sample;  
run;
```

Vi använder data-setet `"work.randomexp"`. Vi beräknar endast medelvärden med `"proc means"`. Vi vill spara dessa medelvärden i `"work.exponential_means"` och vi döper variabeln till `"xbar"` ( $\bar{x}$ ). Eftersom våra slumpstal ligger i variabeln `"x"` skriver vi `"var x"`. Vi skriver `"by sample"` för att vi vill beräkna ett medelvärde för varje stickprov.

Vi ska nu standardisera medelvärdena. Vi gör det genom att tillämpa

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \sim N(0, 1).$$

Om täthetsfunktionen för exponentialfördelningen ser ut så här

$$\begin{aligned} f(x) &= \frac{1}{\lambda} e^{-\frac{x}{\lambda}} \text{ för alla } x \geq 0 \\ f(x) &= 0 \text{ för alla } x < 0 \end{aligned}$$

är  $\mu = \lambda$  och  $\sigma^2 = \lambda^2$ . I vårt fall ger det att  $\mu = 1$  och  $\sigma^2 = \sigma = 1$ . Vi skriver koden

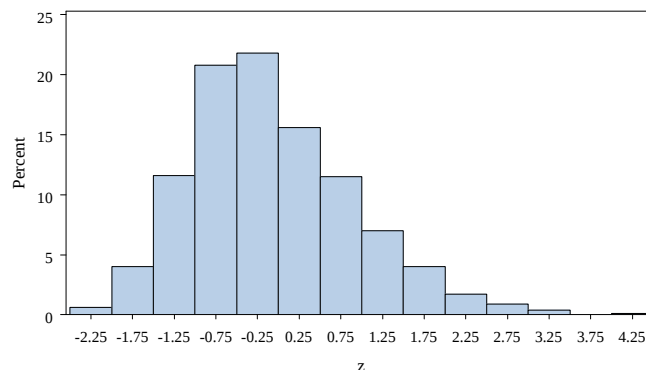
```
data work.exponential_means;
set work.exponential_means;
z = (xbar - 1)/(1/sqrt(10));
run;
```

Vi använder återigen `"data....set..."` för att modifiera variabeln i data-setet. Vi vill nu definiera en variabel `"z"`, som ska vara standard normalfördelad. Vi tittar på data-setet med `"proc print"`.

Nu är det dags att rita ett histogram för vår variabel `"z"`. Vi skriver koden:

```
proc univariate data=exponential_means;
histogram z;
run;
```

Och får diagrammet



Ser det ut som en standard normalfördelning?

## Uppgifter

### Basuppgifter

1. Gör om exempel "Beräkna konfidensintervall och täckningsgrad" och använd ett annat slumpvalsfrö samt ändra antalet stickprov till 1000. Hur många konfidensintervall innehåller 0.1?
2. Rita en exponentialfördelning med valfritt  $\lambda$ .
3. Använd koden i exemplet "illustrera centrala gränsvärdessatsen". Ändra stickprovsstorleken så att ni ser att fördelningen konvergerar mot standardnormalfördelningen. Vilken variabel är det som behöver bli större, "obs" eller "sample"?