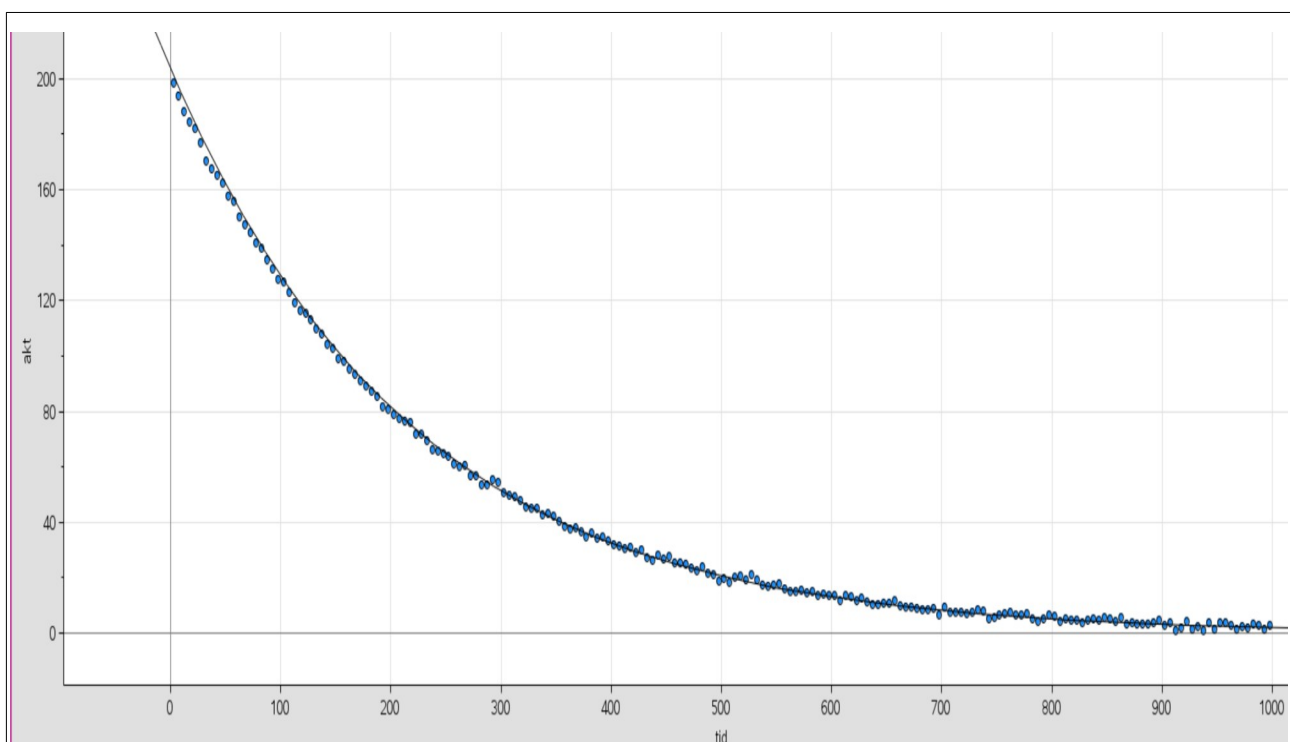


Monte Carlo-simuleringer og statistiske hændelser i fysik og matematik



Figur 1: Monte Carlo-simuleret henfald af Ba^*-137 . Bemærk at spredningen omkring fittet bliver større og større, des mindre tællel.

Indholdsfortegnelse

1 Indhold.....	3
1.1 Historie (von Neumann/Ulam).....	3
1.2 Beskrivelse af metoden.....	3
2 Eksempler fra matematikken.....	3
2.1 Undersøgelse af terninger.....	3
2.1.1.1 Øvelse.....	3
2.1.2 Eksempel på kode.....	3
2.1.3 Resultater fra programmet i afsnit 2.1.2.....	4
2.1.3.1 Øvelse.....	4
2.2 Bestemmelse af tallet π	4
2.2.1.1 Øvelse.....	4
2.2.2 Eksempel på kode til bestemmelse af tallet π	4
2.2.3 Resultater fra programmet i 2.1.2.....	5
2.3 Løsning af integraler.....	5
3 Generel teori for løsning af integraler.....	5
3.1 Statistiske forudsætninger.....	5
3.2 Beregning af integralet.....	6
3.2.1 Eksempel på $f(x)=\sin(x)$ uden bestemmelse af spredning.....	6
3.2.2 Eksempel på $f(x)=\sin(x)$ med bestemmelse af spredning.....	7
3.2.2.1 Øvelse.....	7
3.2.3 Resultater ($a = 0$ og $b = \pi$).....	7
3.2.3.1 Øvelse.....	7
3.2.4 Monte Carlo integrations effektivitet.....	8
3.3 Integraler i 3 dimensioner.....	8
4 Eksempler fra fysikken.....	8
4.1 Eksempel – Plancks strålingslov.....	8
4.2 Radioaktivitet.....	9
4.2.1.1 Øvelse.....	10
4.2.1.2 Øvelse.....	11
4.2.2 Poission- kontra Normalfordeling.....	11
4.2.3 Opgave.....	11
5 Referencer.....	12

1 Indhold

1.1 Historie (von Neumann/Ulam)

Monte Carlo-simuleringer blev udviklet under og især efter 2. verdenskrig, og i dag vil de fleste nok mene, at det var Stanislaw Ulam, der står fadder til metoden. Han og John von Neumann begyndte at arbejde seriøst med simuleringer i 1946 i arbejdet med at forbedre kernevåben. [1]

1.2 Beskrivelse af metoden

Metoden går ud på at benytte tilfældige tal som uafhængige variable i en matematisk model af en problemstilling, og dermed undersøge modellens forudsigelser.

I videoen i [2] er vist, hvordan man ved hjælp af tilfældigheder kan bestemme et mål for tallet π . Se den video, før du læser videre.

2 Eksempler fra matematikken

2.1 Undersøgelse af terninger

En terningespiller vil gerne vide hvad sandsynligheden er for at få en bestemt værdi ved kast af to terninger. Spilleren kan kaste de to terninger utallige gange og måle hvor mange gange de bestemte udfald fremkommer. Et problem kan dog være at de anvendte terninger er såkaldt *uærlige*, dvs. de ikke har samme sandsynlighed for alle udfald. Et andet problem er, at sandsynligheder kun er gode, hvis et forsøg udføres mange gange. [3] Her kan det være nyttigt at lave et lille computerprogram, der simulerer terningekast.

2.1.1.1 Øvelse

Et eksempel på et BASIC-program er vist i næste afsnit. Prøv at skrive programmet ind i nSpire og kød det. Tjek tallene i tabellen på næste side.

2.1.2 Eksempel på kode

```
Define mcterning(n,værdi,seed)=
Prgm
:optæl:=0
:RandSeed seed
:For i,1,n
:  udfald:=randInt(1,6)+randInt(1,6)
:  If udfald=værdi Then
:    optæl:=optæl+1
:  EndIf
:EndFor
:Disp "Antal slag med værdien ",værdi," er ",optæl," ud af ",n," kast."
:Disp "Procentsatsen af slag med ovenstående værdi er ",format(((optæl)/(n))*100,"F2")," %."
:EndPrgm
```

2.1.3 Resultater fra programmet i afsnit 2.1.2

Værdi ↓. $n \rightarrow$	100	10^3	10^5	10^7	Teoretisk værdi
6	9,00 %	12,80 %	13,85 %	13,88 %	13,9 %
7	16,00 %	15,20 %	16,72 %	16,66 %	16,67 %
8	21,00 %	15,90 %	13,87 %	13,89 %	13,89 %
9	10,00 %	11,00 %	11,16 %	11,11 %	11,11 %

2.1.3.1 Øvelse

Prøv at udvide programmet fra afsnit 2.1.2 til at simulere kast n terninger pr. kast. (I eksemplet ovenfor var $n = 2$; nu skal det være et selvvalgt antal.)

2.2 Bestemmelse af tallet π

Tallet π er et reelt tal, som i en lommeregner er tilnærmet til tallet $\pi \approx 3,1415926535897932384626433832795\dots$

Teorien bag programmet er gennemgået i [2]. Kort opridset her kan vi sige, at hvis vi fordeler N punkter i en kasse med sidelængde 1, så vil hvert punkt dække et areal $a = 1^2/N = 1/N$. Hvis vi indtegner en kvartcirkel med radius 1 i kassen, så vil denne kvartcirkel have arealet $A = \pi \cdot 1^2/4 = \pi/4$.

Dvs. at hvis vi spreder N punkter tilfældigt rundt i kassen, så vil antallet af punkter, der ligger inden for radius svare til cirkelns areal. Dermed kan π bestemmes. Herunder er vist et program, der bestemmer cifrene i tallet.

2.2.1.1 Øvelse

Indtast nedenstående program, og diskuter med din sidemand, hvad programmet gør. Undersøg også tallene i tabellen i afsnit 2.2.3.

2.2.2 Eksempel på kode til bestemmelse af tallet π

```

Define mcpi(n,seed)=
Prgm
:DelVar estpi,a
:a:=0
:RandSeed seed
:For i,1,n
: x:=rand()
: y:=rand()
: l:=x^(2)+y^(2)
:If l<=1 Then
: a:=a+1
:EndIf
:EndFor
: estpi:=4*((a)/(n))
:Disp "π = ",estpi," ved ",n," iterationer."
:EndPrgm

```

2.2.3 Resultater fra programmet i 2.1.2

n	10	100	10^3	10^6	10^7	10^8
Værdi for π	3,2	3,28	3,136	3,14218	3,14172	3,14149
$\Delta\pi_{\text{rel}}$	1,9 %	4,4 %	1,8 %	0,019 %	0,004 %	0,0033 %

2.3 Løsning af integraler

Man kan også bruge Monte Carlo-metoden til at løse indviklede integraler. Se videoen i kilden før du læser videre. [4]

3 Generel teori for løsning af integraler

Vi vil løse integralet, F , af $f(x)$ i intervallet $[a, b]$. Dvs. vi vil løse

$$F = \int_a^b f(x) dx \quad (1)$$

3.1 Statistiske forudsætninger

Antag en fordelingsfunktion

$$t(x) = \frac{1}{b-a}, \text{ for } x \in [a, b] \text{ og } t(x) = 0 \text{ udenfor intervallet.} \quad (2)$$

Dermed kan middelværdi, varians og spredningsfunktionerne fra statistikken bruges. Dvs. vi kan skrive

$$\langle x \rangle = \int_a^b x \cdot t(x) dx \quad (3)$$

Det aritmetiske gennemsnit for et diskret datasæt kan skrives som

$$\langle x \rangle = \frac{1}{N} \cdot \sum_{i=1}^N x_i \quad (4)$$

Hvis et målesæt er ”stort nok” vil resultatet i (4) gå imod resultatet i (3).

Formel (3) gælder også for en funktion, dvs. vi kan skrive

$$\langle f(x) \rangle = \int_a^b f(x) \cdot t(x) dx \quad (5)$$

Tilsvarende kan vi skrive for variansen, σ^2

$$\sigma^2 = \int_a^b f(x)^2 - \langle f(x) \rangle^2 dx \quad (6)$$

For et diskret sæt af målinger kan vi skrive

$$\sigma_N^2 = \frac{1}{N} \cdot \sum_{i=1}^N (f(x_i) - \langle f \rangle)^2 = \langle f^2 \rangle - \langle f \rangle^2 \quad (7)$$

Spredningen på middelværdien, $\sigma(\langle f \rangle)$, er så

$$\sigma(\langle f \rangle) = \frac{\sigma_N}{\sqrt{N}} \quad (8)$$

3.2 Beregning af integralet

Vi indsætter nu fordelingsfunktionen (2) i formelen for middelværdien (5) og vi får dermed

$$\langle f \rangle = \frac{1}{b-a} \cdot \int_a^b f(x) dx \quad (9)$$

Hvis vi beregner et (stort) antal diskrete funktionsværdier for $f(x)$, kan vi beregne det aritmetiske gennemsnit ud fra formelen

$$\langle f \rangle = \frac{1}{N} \cdot \sum_{i=1}^N f(x_i) \quad (10)$$

Vi sætter udtrykkene i (9) og (10) lig med hinanden og isolerer integralet, og det giver

$$F = \int_a^b f(x) dx \approx \frac{b-a}{N} \cdot \sum_{i=1}^N f(x_i) \quad (11)$$

Vi er også interesseret i spredningen på F , og den er

$$\sigma(F) = (b-a) \cdot \sigma(\langle f \rangle) = \frac{(b-a) \cdot \sigma}{\sqrt{N}} \quad (12)$$

Vi kan sammenfatte opskriften på Monte Carlo-integration som følger:

- Opskriv funktionen $f(x)$.
- Beregn et stort antal funktionsværdier $f(x_i)$ i intervallet $[a, b]$.
- Beregn et tilnærmet mål for integralet ved at bruge (11).
- Undervejs beregnes $f(x_i)^2$ så spredningen kan beregnes ved hjælp af (7), (8) og (12).

Herunder er et kort eksempel på at beregne integralet ved hjælp af nSpires sum-funktion, men prisen er så, at man ikke får spredningen med.

3.2.1 Eksempel på $f(x)=\sin(x)$ uden bestemmelse af spredning

```
Define mcintegral(a,b,n,seed)=
Prgm
:RandSeed seed
:f:=((b-a)/(n))*sum(sin(a+(b-a)*rand()),i,1,n)
:Disp "Integralet af sin(x) fra ",a," til ",b," er ",f
:EndPrgm
```

Ovenfor er udnyttet at nSpire har indbyggede sum-funktioner, så koden kan skrives hurtigt ned.

Herunder er et eksempel på beregning af integral og spredning, og programmet kan også nemt oversættes til andre programmeringssprog.

3.2.2 Eksempel på $f(x)=\sin(x)$ med bestemmelse af spredning

```
Define mcintegral(a,b,n,seed)=
Prgm
:RandSeed seed
:y:=0
:y2:=0
:s:=0
:s2:=0
: For i,1,n
:  x:=a+(b-a)*rand()
:  f:=sin(x)
:  y:=y+f
:  y2:=y2+f^2
: EndFor
:middelværdi:=(y)/(n)
:middelf2:=(y2)/(n)
:integralet:=(b-a)*middelværdi
:
:s2:=middelf2-middelværdi^2
:s:=(b-a)*(((s2)/(n)))^0.5
:Disp "Integralet af sin(x) fra ",a," til ",b," er ",integralet
:Disp "Spredningen er ",s
:EndPrgm
```

3.2.2.1 Øvelse

Indtast programmet ovenfor og diskuter med din sidemand, hvad programmet gør. Undersøg dernæst resultattabellen, som er vist i afsnit 3.2.3.

3.2.3 Resultater ($a = 0$ og $b = \pi$)

N	100	1000	10^4	10^5	10^7
Resultat	1,949	2,0105	1,99685	1,99622	2,000
$\sigma(\langle f \rangle)$	0,0974	0,03	$9,67 \cdot 10^{-3}$	$3,06 \cdot 10^{-3}$	$3,06 \cdot 10^{-4}$
Relativ afvigelse (fra sand værdi)	7,655 %	0,185 %	0,8645 %	0,0325 %	0,0275 %

3.2.3.1 Øvelse

Udvid programmet i eksempel 3.2.2 med en funktions-rutine, sådan at man hurtigt kan ændre programmet til at løse integralet for en anden funktion. Hvis du ikke før har prøvet at lave funktioner i nSpire, så er der et eksempel på en simpel funktion herunder.

```
Define min_funktion(n)=
Func
:Return n*π
:EndFunc
```

3.2.4 Monte Carlo integrations effektivitet

Andre integrationsmetoders fejl aftager hurtigere mht. Antal beregninger, N , når man løser integraler af 1 variabel. Men arbejder man med funktioner af flere variable, aftager fejlen væsentligt langsommere for andre metoder, mens det forbliver som $N^{-1/2}$ for Monte Carlo-metoden. Det viser sig, at hvis antallet af dimensioner overstiger 4 er MC-integration hurtigere end trapez-metoden, og hvis antallet af dimensioner overstiger 8, så er MC-integration hurtigere end Simpsons metode.

3.3 Integraler i 3 dimensioner

Hvis ens funktion F afhænger af flere, f.eks. 3 variable, kan den skrives som

$$F = \int_{a_x}^{b_x} \int_{a_y}^{b_y} \int_{a_z}^{b_z} f(x, y, z) dx dy dz \quad (13)$$

Dermed bliver den aritmetiske middelværdi

$$\langle f \rangle = \frac{1}{N} \cdot \sum_{i=1}^N f(x_i, y_i, z_i) \quad (14)$$

En approksimeret værdi for F bliver så

$$F \approx (b_x - a_x) \cdot (b_y - a_y) \cdot (b_z - a_z) \cdot \langle f \rangle \quad (15)$$

$$\sigma_N^2 = \langle f^2 \rangle - \langle f \rangle^2 \quad (16)$$

$$\sigma(\langle f \rangle) = \frac{\sigma_N}{\sqrt{N}} \quad (17)$$

Og endelig er

$$\sigma(F) = \sigma(\langle f \rangle) \cdot \prod_{i=x}^z (b_i - a_i) \quad (18)$$

Man kan læse mere i J. Puls [5]. Ovenstående kan generaliseres til et vilkårligt antal dimensioner.

4 Eksempler fra fysikken

4.1 Eksempel – Plancks strålingslov

Hvis man vil integrere plancklovens integral

$$\int_0^{\infty} \frac{x^3}{e^x - 1} dx$$

kan man benytte nedenstående eksempel.

```
Define mcintegral(a,b,n,seed)=
Prgm
:RandSeed seed
:y:=0
:y2:=0
```

```

:s:=0
:s2:=0
: For i,1,n
:  x:=a+(b-a)*rand()
:  f:=funkt(x)
:  y:=y+f
:  y2:=y2+f^2
: EndFor
:middelværdi:=(y)/(n)
:middelf2:=((y2)/(n))
:integralet:=(b-a)*middelværdi
:
:s2:=middelf2-middelværdi^2
:s:=(b-a)*(((s2)/(n)))^0.5
:Disp "Integralet af funk(x) fra ",a," til ",b," er ",integralet
:Disp "Spredningen er ",s
:EndPrgm

```

```

Define funk(x)=
Func
:If x=0 Then
:  Return 0
:Else
:  Return ((x^(3))/(e^(x)-1))
:EndIf
:EndFunc

```

Med $N = 10^7$ får man integralet til at være $6,496 \pm 0,039$. Facit skal være $\pi^4/15 = 6,4939$. Men det tager ret lang tid at regne integralet ud, så eksemplet viser ikke det kraftfulde i Monte Carlo-simuleringer.

4.2 Radioaktivitet

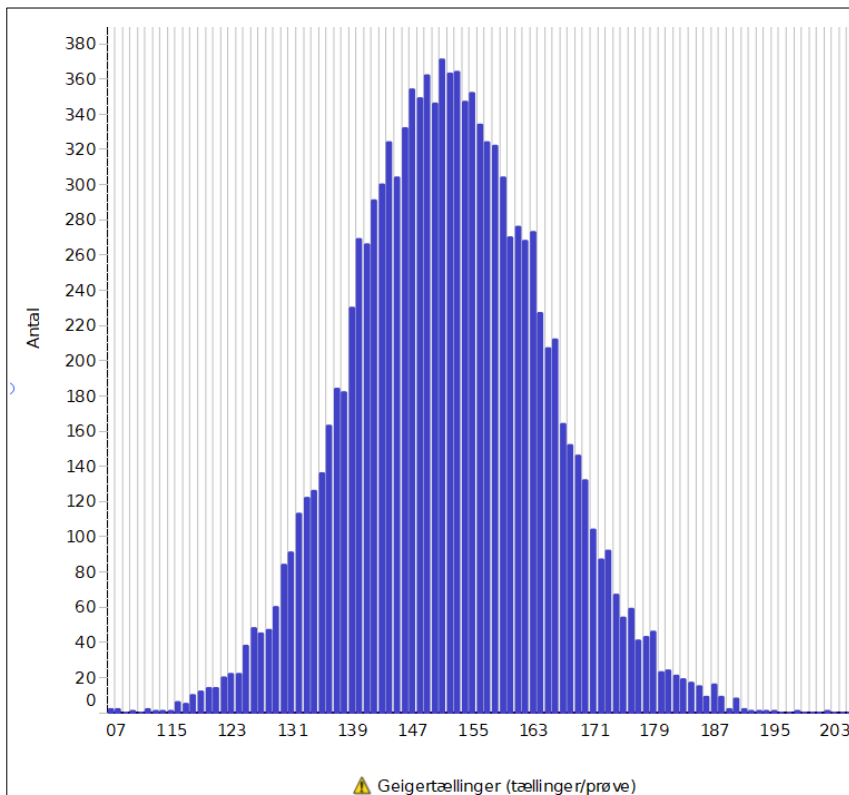
En radioaktiv kilde har aktiviteten A . Hvis kilden har en "lang" halveringstid i forhold til det tidsrum, man ønsker at måle over, kan vi kalde aktiviteten for A_0 . Aktivitetsloven er kendt som

$$A(t) = A_0 \cdot \exp(-k \cdot t) \quad (19)$$

Ovenfor er k sandsynligheden for henfald pr. kerne pr s.

Når man måler aktiviteten fra en radioaktiv kilde, viser henfaldets statistiske natur sig hurtigt ret tydeligt. F. eks. vil man ikke måle en konstant aktivitet fra en kilde, selvom halveringstiden er væsentligt længere end måleperioden. Det fænomen er vist på figur 2.

I eksperimentet skal man måle tælle-tallene, ΔN , i et vist tidsrum, Δt , for at man kan få



Henfaldet af Cs-137. Der er målt i intervaller à 10 s.

Figur 2: Måling på Cs-137 med Pasco måleudstyr. Der er målt i 5 s-intervaller. I alt er der foretaget 11474 målinger.

store nok tælleantal. Hvis man kalder den detekterede brøkdel af den stråling, som rammer detektoren til at være η , og hvis man antager kuglesymmetri, dvs. detektoren er anbragt i afstanden, d , fra kilden, ville man kunne beregne aktiviteten ved formlen

$$A(t) \approx \frac{4 \cdot d^2 \cdot \Delta N}{\eta \cdot r^2 \cdot \Delta t} \quad (20)$$

Ovenfor angiver r radius af GM-røret.

4.2.1.1 Øvelse

a) Vis formel (20).

I eksperimentet, der gav dataene til figur 2, var $d = 0,10$ m, $r = 0,02$ m, og $\Delta t = 10$ s. Antag at $\eta = 1,5$ %.

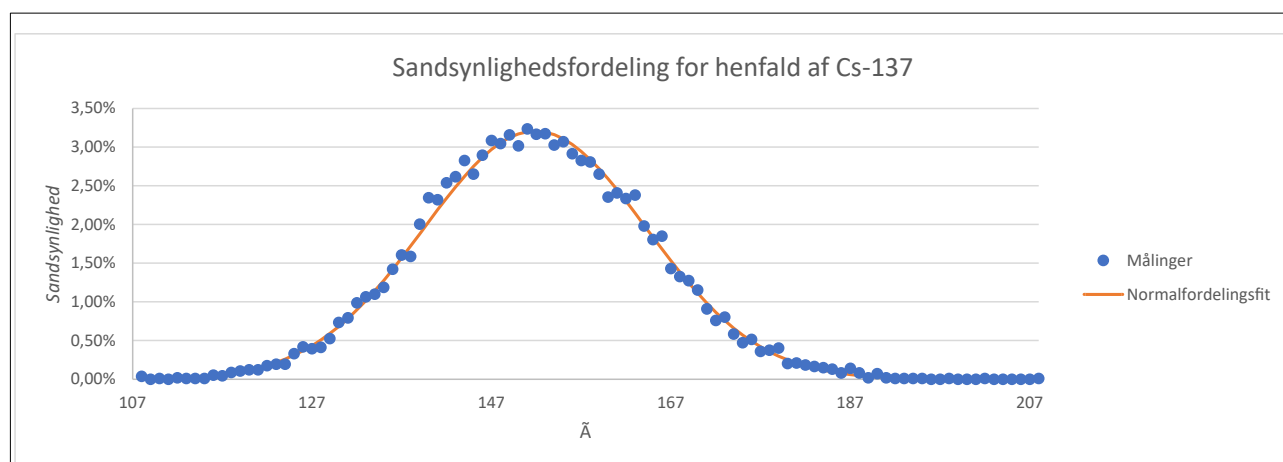
b) Beregn aktiviteten $A(t)$.

Kilden er fra 2013 og målingerne er foretaget i 2023. Halveringstiden for Cs-137 er 30 yr.

c) Beregn A_0 .

-0-

Man ser at figur 2 ligner en normalfordeling vældig meget, og nedenfor er vist et fit til en normalfordeling. RMSE-fejlen er på 0,00078. Dvs. henfaldet beskrives med stor præcision som en normalfordeling.



Figur 3: De blå punkter er målepunkter, og den orange linie er en tilpasset normalfordeling til målingerne. Middelværdien er $\tilde{A} = 152$ og spredningen er $\sigma = 12,4$. Spredningen på middelværdien er 0,12.

Formlerne (4), (7) og (8) er anvendt til at finde målingernes middelværdi, spredning og spredning på middelværdien. Når man bruger dem som startværdier i normalfordelingsfittet, kan en *Data-What If-analyse-målsøgning* i Excel bruges til at minimere RMSE-fejlen.

Sandsynligheden for at måle et tælleantal, $\tilde{A}$, kan beregnes ved hjælp af normalfordelingen, som i vores tilfælde ser ud som følger

$$p(\tilde{A}) \cdot d\tilde{A} = \frac{1}{\sigma \cdot \sqrt{2 \cdot \pi}} \cdot \exp \left[-\frac{1}{2} \cdot \left(\frac{\tilde{A} - \mu}{\sigma} \right)^2 \right] \cdot 1 \quad (21)$$

4.2.1.2 Øvelse

- Beregn $p(110)$, $p(152)$ og $p(164)$ ved at bruge formel (21). Antag den middelværdi og spredning, som er anført i figurteksten til figur 3.
- Sammenlign dine resultater med figur 3.
- I datasættet var der 11474 målinger. Gang dette tal på f. eks. $p(152)$ og find det tilhørende punkt på figur 3.

4.2.2 Poission- kontra Normalfordeling

Man lærer ofte, at poissonfordelinger beskriver radioaktivitet, og her har vi brugt en normalfordeling. Det viser sig dog, at hvis middelværdien er over ca. 20 (med enhed), så beskrives henfaldet også ved hjælp af en normalfordeling med $\sigma = \langle \tilde{A} \rangle^{1/2}$. I analysen, som er vist i figur 3, ser vi at $12,4^2 = 154 \approx \langle \tilde{A} \rangle = 152$.

4.2.3 Opgave

En isotop har en halveringstid på $T_{1/2}$. Der skal beregnes en aktivitetskurve $\tilde{A}(t)$ for isotopen. Der skal foretages n 'målinger' over tidsrummene $\Delta t \sim T_{1/2}$. Tildesymbolet til venstre betyder, at halveringstiden i størrelse er sammenlignelig, men dog større end måletidsrummene. Antag, at aktiviteterne er normalfordelte.

Skriv et nSpire-program, der simulerer en aktivitetskurve. Inputparametre skal være n , Δt , $T_{1/2}$, $\tilde{A}_0$ og σ .

Du behøver ikke vælge en afstand til kilde samt radius af GM-rør, men du skal integrere en simuleret aktivitet $\tilde{A}(t)$ ved Monte Carlo-integration.

5 Referencer

1. https://en.wikipedia.org/wiki/Monte_Carlo_method1.
2. <https://www.youtube.com/watch?v=7ESK5SaP-bc>
3. https://www.ge.com/digital/documentation/meridium/V36000/WebHelp/mergedProjects/WebHelp_Master/mergedProjects/Reliability/Monte_Carlo_Simulations_A_Simple_Example.htm
4. <https://www.youtube.com/watch?v=WAf0rqwAvgg>
5. Joachim Puls: "Computational Methods in Astrophysics." 1st ed, 2005, Ludwigs-Maximilians-Universität München.
6. https://ned.ipac.caltech.edu/level5/Leo/Stats2_2.html