

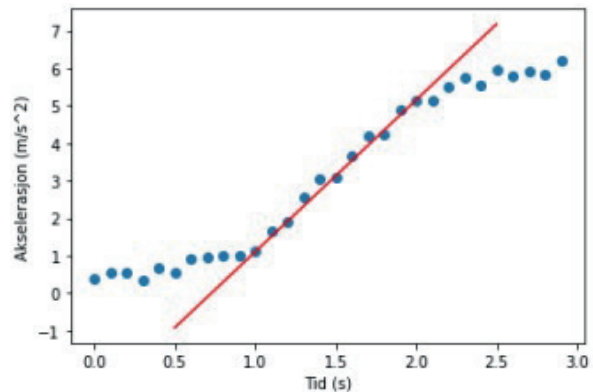
Opplegg 10 – Momentan vekstfart

To former for vekstfart

Vi skiller mellom gjennomsnittlig og momentan vekstfart. Den gjennomsnittlige vekstfarten forteller oss hvor raskt noe øker mellom to x-verdier, mens den momentane vekstfarten forteller oss hvor rask økningen er i et bestemt punkt. Vi begynner med den gjennomsnittlige.

Gjennomsnittlig vekstfart

I følgende kode forutsetter vi at vi allerede har importert et datasett med x- og y-verdier. Vi starter med å plote punktene. Deretter beregner vi den gjennomsnittlige akselerasjonen ved å bruke $a = \Delta x / \Delta y$. I vår figur var $N = 30$, og vi ønsker å beregne den gjennomsnittlige vekstfarten for indeksverdier mellom 10 og 20.



```
scatter(x,y)
xlabel("Tid (s)")
ylabel("Akselerasjon (m/s^2)")

aks_gj = (y[20]-y[10])/(x[20]-x[10])
```

Plotter x- og y-verdiene. Beregner deretter den gjennomsnittlige akselerasjonen.

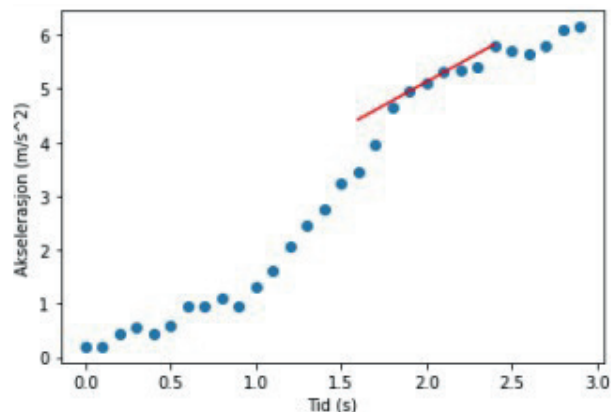
```
def f(x,a,x1,y1):
    return a*(x-x1)+y1

s = linspace(0.5,2.5)
t = f(s,aks_gj,x[10],y[10])
plot(s,t,'r')
```

Bruker ettpunktsformelen til å finne funksjonsuttrykket til den røde linja på figuren. Til sist plotter vi linja.

Momentan vekstfart

Hvis vi vil finne den momentane vekstfarten, gjør vi akkurat som for den gjennomsnittlige, men vi velger bare to punkter som ligger rett i nærheten av hverandre. Dermed får vi et uttrykk for hvordan funksjonsverdiene er i ferd med å endre seg akkurat der. I eksempelet nedenfor har vi valgt indeksverdier 19 og 21 for å se på veksten rundt $i=20$, altså $x=2,0$.



Plotter grafen og beregner gjennomsnittlig akselerasjon for x-verdier i nærheten av hverandre.

```
scatter(x,y)
xlabel("Tid (s)")
ylabel("Akselerasjon (m/s^2)")
aks_gj = (y[21]-y[19])/(x[21]-x[19])
```

Bruker ettpunktsformelen og plotter den røde linja.

```
def f(x,a,x1,y1):
    return a*(x-x1)+y1
s = linspace(1.6,2.4)
t = f(s,aks_gj,x[19],y[19])
plot(s,t,'r')
```

Lag en utskytningsmekanisme

- mål tid og akselerasjon med micro:bit og lagre til fil

Oppgave

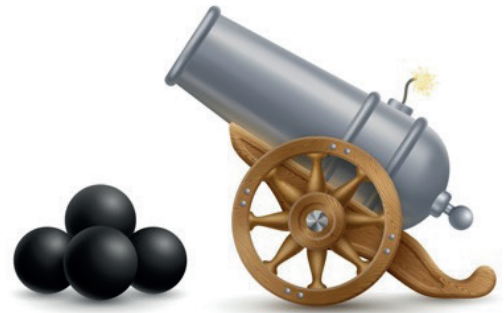
Lag en utskytningsmekanisme som skyter en micro:bit med støtdemping rundt slik at «micro:bit-kula» får størst mulig rykk (akselerasjonsendring). Micro:biten skal måle akselerasjonen, og disse dataene skal dere bruke til å beregne den største momentane vekstfarten for akselerasjonen (rykket) micro:biten hadde på flygeturen.

Fase 1: Gjennomfør informasjonsinnhenting for å få idéer. Hva slags utskytningsmekanismer finnes?

Fase 2: Idémyldre og planlegge

Det er viktig at dere er åpne for alle slags ideer og ikke er for kritiske, da kan nyttige forslag bli avfeid for tidlig.

1. Tenk selv først og tegn gjerne skisser.
2. Forklar ideen din for de andre på gruppa.
3. Hele gruppa diskuterer de ulike ideene, og lager en felles plan for bygging.



Fase 3: Gjennomføring

- Bygg utskytningsmekanismen
- Lag programmet for å måle akselerasjonen med micro:biten og pakk micro:biten inn i støtdempende materiale.

Fase 4: Test om utskytningsmekanismen og programmene virker som planlagt.

Fase 5: Sammenlign resultatene med andre i klassen.

- Fikk noen andre større akselerasjon?
- Hvorfor det?
- Kan dere bruke noe av det samme på deres utskytningsmekanisme for å få den mer effektiv?

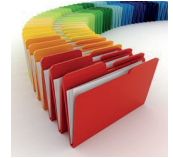
Fase 6: Gå tilbake til de andre fasene for å gjøre de planlagte forbedringene.

Fase 7: Gjennomfør de siste målingene, og dokumenter prosjektet på en valgfri måte.

Programmeringsoppgaver

1. Lag en regresjonsmodell for akselerasjonen som funksjon av tiden og lag et plott. Se tips på side 4 og 5. Diskuter hva som kan være gyldighetsområde for modellen.
2. Deriver modellen fra 1. og plott resultatet. Hvordan vil du beskrive akselerasjonsendringen ifølge denne modellen?
3. Bruk de målte verdiene for tid og akselerasjon, og lag et program som finner momentan vekstfart numerisk. Lag en graf av akselerasjonsendringen som en funksjon av tiden. Se tips på side 3. Ser grafen ut slik dere forventet? Når er endringen størst?
4. Sammenlign grafen fra 3. med den deriverte regresjonsmodellen. Hvilken er mest nøyaktig?
5. Ser det ut til at akselerasjonsendringen minker med tiden? Hvordan henger dette sammen med utskytningsmekanismen?

Skrive til fil på micro:bit med Python Mu



I slutten av kapittel 9 brukte vi datasett i form av tekstfiler for å lage matematiske modeller. Hadde det ikke vært kult om vi kunne fått micro:biten til å lage slike filer av sine målinger? Da kan vi bruke micro:biten til å samle inn data over en viss periode, uten å være koblet til en datamaskin. Datasettet blir lagret i micro:biten som en .txt-fil, og vi kan legge den over i datamaskinen, før vi plottes resultatene eller lager en matematisk modell med Python.

Det aller første vi må gjøre, er å opprette en tekstfil, det vil si en fil som slutter på .txt, og det er enklest å gjøre i Mu. Lagre den samme i samme mappe som resten av dine Python Mu-filer.

Deretter må den fila du har laget, kopieres over på micro:biten. Dette gjør du på samme måten som for sonaren.

```
with open ('filnavn.txt', 'w') as fil:
with open ('filnavn.txt', 'r') as fil:
fil.write(y+'\n')
```

↓

Skriver variabelen y til fila vår. '+\n' gjør at det blir et linjeskift etter hvert tall, og det må vi ha for at python skal kunne lese filene. Må stå med innrykk.

```
z = fil.read()
```

↓

Leser det som står i fila vår, og legger det inn i variabel z. Må stå med innrykk.

Åpner fila som heter filnavn.txt og tildeler den til et objekt som vi kaller fil. Siden vi har 'w' (write) i kommandoen gjøres det klar til å skrive inn i fila.

Tilsvarende som linja over, men siden vi har 'r' (read) i kommandoen gjøres det klar til å lese fila. Det er valgfritt å ta med 'r'.

```
y = str(x)
```

→

Gjør om variabelen x til en streng (tekstverdi). Må stå med innrykk.

```
print(z)
```

→

Skriver variabelen z til skjermen på PC. Må IKKE stå med innrykk.

Da gjenstår det å lage selve programmet. Under er et eksempel der micro:biten måler temperaturen og legger målingen i en .txt-fil ved navn datasett.txt. Det kan være greit å kunne sjekke hvordan datasett.txt-fila ser ut uten å overføre til datamaskinen og åpne den, derfor er dette også med i puslespill-oppgaven.

```
Puslespill-oppgave
print(innhold)      t = temperature()
while not button_a.is_pressed():
sleep(1000)         fil.write(str(t)+'\n')
with open('datasett.txt') as fil:
with open('datasett.txt', 'w') as fil:
innhold = fil.read()  from microbit import *
```

Ekstraoppgave

Lag et program som bruker en sensor du kobler på micro:biten for å samle inn data som du lagrer i en fil på micro:biten. Hva må du forandre på? Hva må du forandre i programmet for å lage en fil med to kolonner der den første kolonnen er hvor lang tid det har gått før målingen er gjort?