



**Virtual Instrumentering:
Matlab Introduktion 2**

Samuel Alberg Thrysoe, PhD, PostDoc,
Kontakt info: Email: sat@iha.dk, Tlf: +45 25533552



Repetition

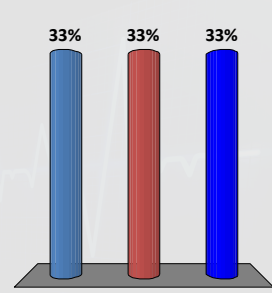
Hvad giver denne kommando:
 $x = 3:2:9$

1. $x =$
3 2 9



2. $x =$
3 5 7 9

3. $x =$
3 4 5 6 7 8 9



| Option | Percentage |
|--------|------------|
| 1 | 33% |
| 2 | 33% |
| 3 | 33% |

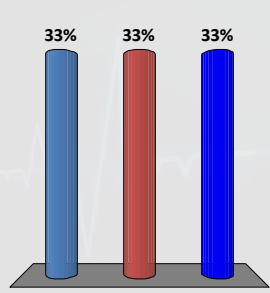
Hvad giver denne kommando:
 $x = [1\ 2\ 3;4\ 5\ 6;7\ 8\ 9]$

1. $x =$
1 2 3 4 5 6 7 8 9



2. $x =$
123 456 789

3. $x =$
1 2 3
4 5 6
7 8 9



| Option | Percentage |
|--------|------------|
| 1 | 33% |
| 2 | 33% |
| 3 | 33% |

Hvad giver denne kommando:
 $A(2,3)$

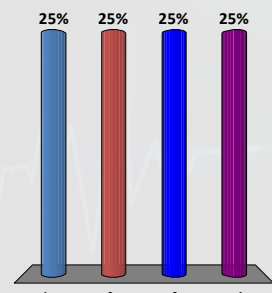
| | | | |
|-----|---|---|---|
| A = | 3 | 5 | 3 |
| | 6 | 8 | 2 |
| | 2 | 7 | 3 |

1. 2

2. 7



3. 3

4. 8



| Option | Percentage |
|--------|------------|
| 1 | 25% |
| 2 | 25% |
| 3 | 25% |
| 4 | 25% |

Hvad giver denne kommando:
 $A(1,:)$

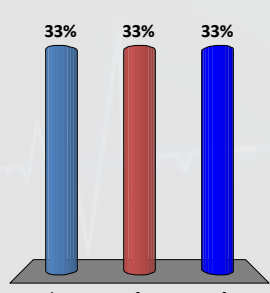



| | | | |
|-----|---|---|---|
| A = | 3 | 5 | 3 |
| | 6 | 8 | 2 |
| | 2 | 7 | 3 |

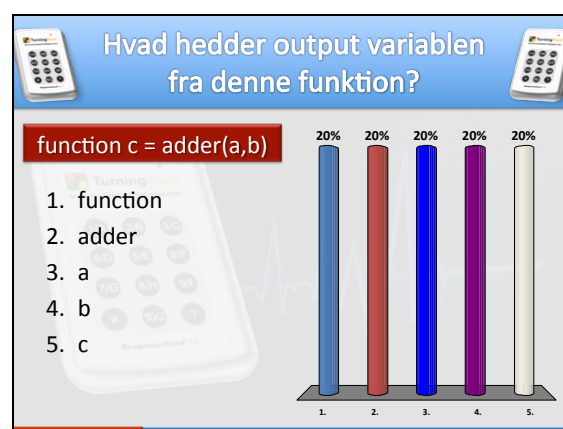
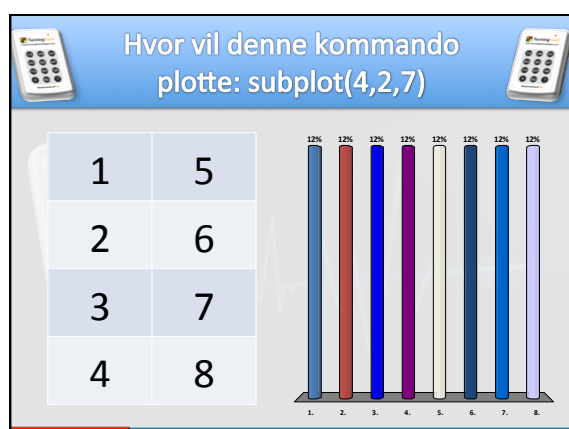
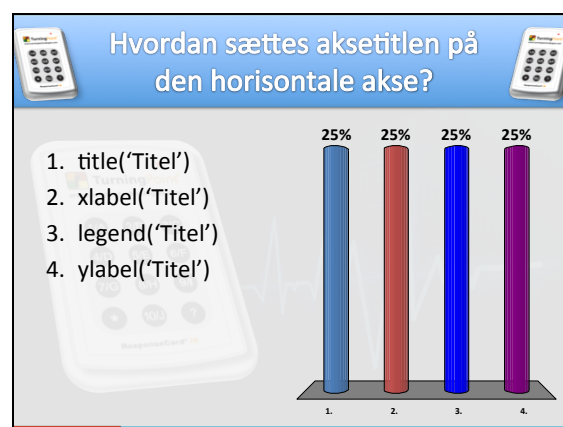
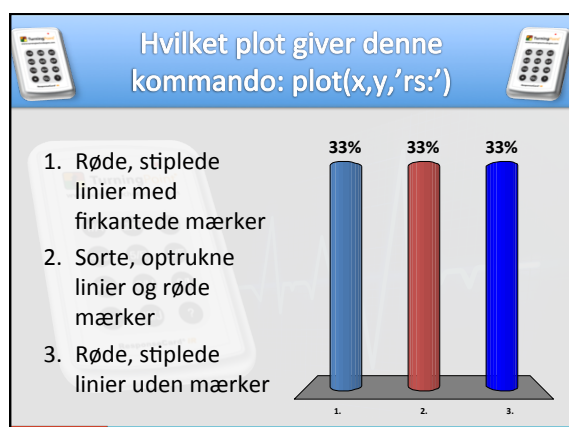
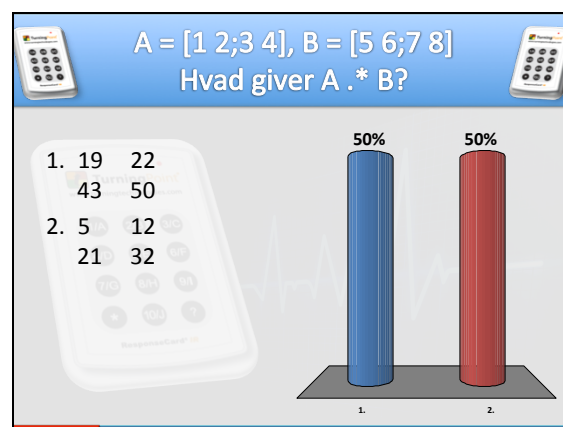
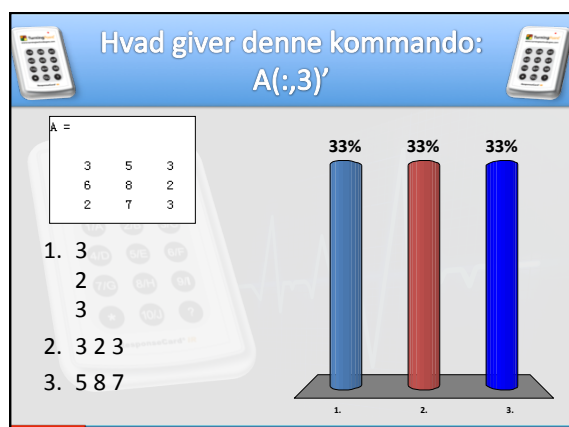
1. 6 8 2

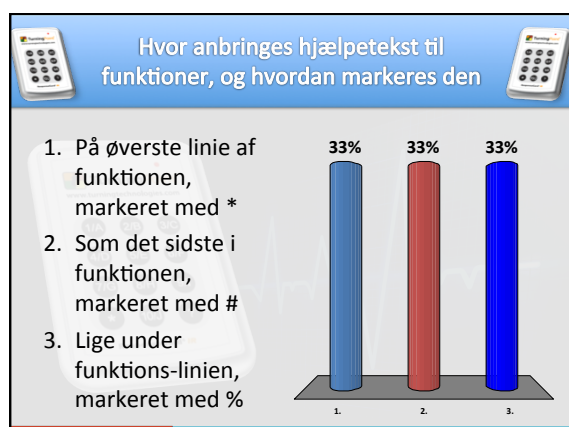
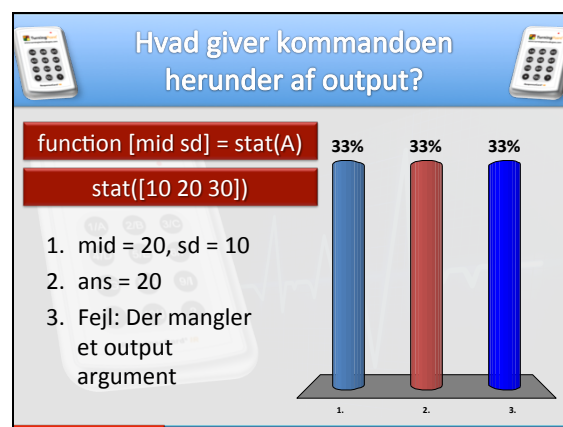
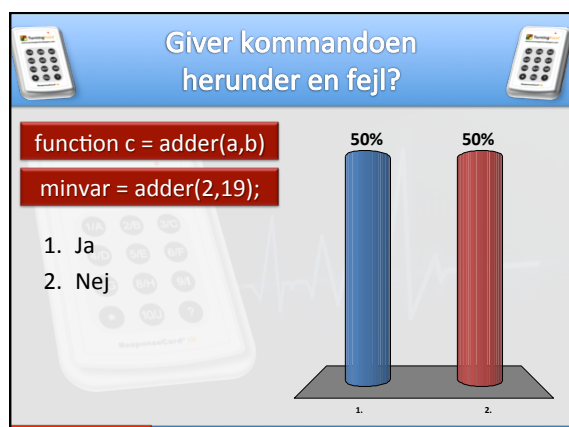
2. 2 7 3

3. 3 5 3



| Option | Percentage |
|--------|------------|
| 1 | 33% |
| 2 | 33% |
| 3 | 33% |





Funktioner

```
function [mid sd] = statistik(A)
% Denne funktion kan levere middelværdi
% og standard deviation fra en input-
% matrix
% Funktionskald:
% [mid sd] = stat(A)
% hvor:
%   mid = middelværdi
%   sd  = standard deviation
%   A   = input-matrix
mid = mean(A);
sd  = std(A);
```



Find Syntaks

- `find(a>90)`

```
ans =
7
16
25
35
45
```

- `[r c] = find(a>90)`

```
r =
7
6
5
5
5
c =
1
2
3
4
5
```

| | 1 | 2 | 3 | 4 | 5 |
|----|----|----|----|----|----|
| 1 | 28 | 76 | 85 | 36 | 8 |
| 2 | 68 | 26 | 26 | 84 | 6 |
| 3 | 66 | 51 | 82 | 59 | 54 |
| 4 | 17 | 70 | 25 | 55 | 78 |
| 5 | 12 | 90 | 93 | 92 | 94 |
| 6 | 50 | 96 | 35 | 29 | 13 |
| 7 | 96 | 55 | 20 | 76 | 57 |
| 8 | 35 | 14 | 26 | 76 | 47 |
| 9 | 59 | 15 | 62 | 39 | 2 |
| 10 | 23 | 26 | 48 | 57 | 34 |

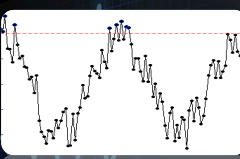
Find på billeder

- `idx=find(im>200)`
- `idx=find(im>m+2*sd)`



8.1.1 - Plot limit øvelse

- Find funktionsværdier over et threshold
 - Cosinuskurve med støj
 - `cos =` værdier mellem 0 og 1
 - `rand =` værdier mellem 0 og 1
 - `cos + rand = 0` til 2
 - Find værdier over 1.5
 - Plot 3 kurver i en:
 - Selve kurven
 - Overstigende punkter
 - Grænseværdi



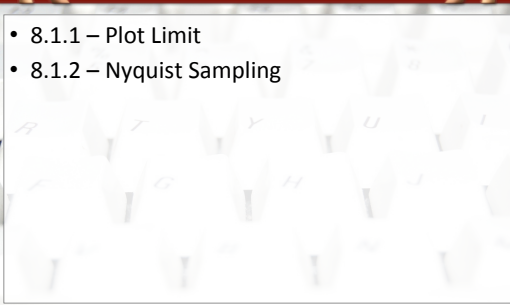
8.1.2 – Nyquist Sampling

- Nyquist-Shannon samplingsteorem
 - Plot sinusurve fra 0-1 med 1000 punkter med brugerbestemt frekvens.
 - Samples med F_s
 - Dette er lig med at tage $1000 / F_s$ antal skridt
 - Forklaring på tavle



8.1 – Plot Repetition

- 8.1.1 – Plot Limit
- 8.1.2 – Nyquist Sampling



Billedbehandling i Matlab



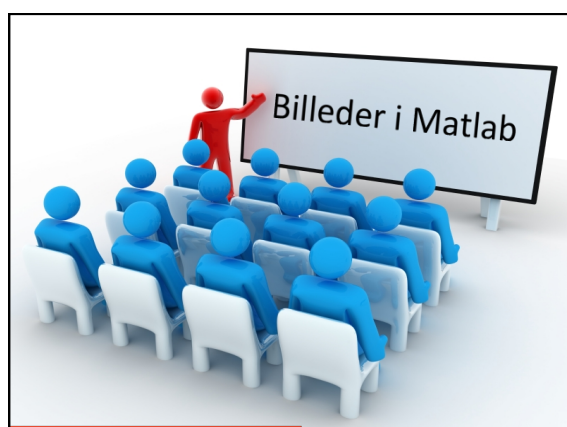
Billede-indlæsning i Matlab

- Matlab kan indlæse
 - jpg
 - gif
 - png
 - ...
- Med funktionen
 - `imread('filnavn')`
 - Ex:
 - `im = imread('billede.jpg');`

Husk semikolon – der kommer store datamængder ud ved denne funktion!

Billeder i Matlab

- Dette giver en ny matrix: `im`
 - Dimensioner
 - Højde x bredde x 3
 - 3-tallet indikerer, der er tale om et farvebillede, som består af rød, grøn og blå
 - Dette udnyttes i senere øvelse
- Billedet kan plottes med `imshow`:
 - `imshow(im);`



Øvelse: Farveblindhed

- Opbyg funktion til at indlæse farvebillede og erstatte grøn med blå
 - Lav først en funktion til at
 - Indlæse et billede
 - Returnere billedet
 - Returnere den røde, grønne og blå kanal som separate billeder

Øvelse: Farveblindhed

- Lav dernæst funktion til at erstatte grøn med blå
 - Find pixels, hvor den grønne kanal overstiger blå og rød med en brugerdefineret tærskelværdi
 - NB: Gul på RGB skalaen er defineret som rød + grøn
 - I kan altså ikke bare bruge den grønne kanal
 - Sæt disse til at være blå

8.2.1 – Farveblindhed

Dicom-billeder

- Medicinsk billedformat: Dicom
 - Indeholder såvel
 - Billedeværdier
 - Information
 - Scanner, sekvens, scannersted, patient navn, id osv.
- Kan indlæses i Matlab med
 - `im = dicomread('filnavn.dcm');`
 - `info = dicominfo('filnavn.dcm');`

Visning af Dicom-billeder

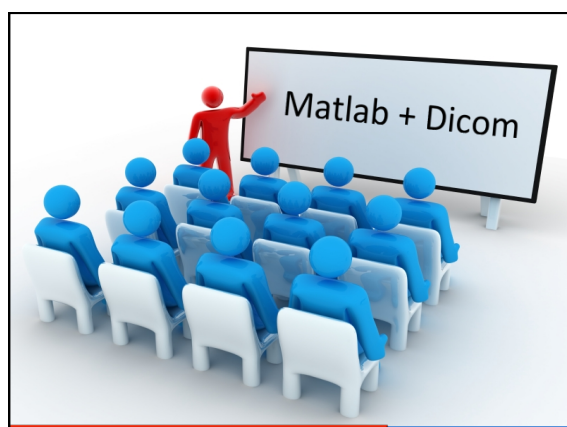
- Kan plottes på sædvanlig vis
 - `imshow(im)`
- MEN:
 - `imshow` antager altid, at billedet løber fra nul til det maksimalt mulige
 - Hvis datatypen fx er `uint16`, er maksimum $2^{16} = \text{ca. } 65000$
 - Hvis billedets max-værdi er fx 700, bliver hele billedet sort med `imshow`

Løsning

- Funktionen `imagesc` plotter et skaleret billede
 - MEN:
 - `imshow` plotter automatisk i det rette størrelsesforhold og uden akser
 - Det gør `imagesc` ikke...
 - Kommandoen
 - `axis equal off`
 - Sætter størrelsesforholdet til at være korrekt
 - Slukker for akserne

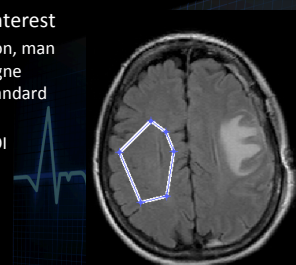
Alternativ løsning

- Skaler billedet til at lægge indenfor 0 til 1:
 - `im = double(dicomread('filnavn.dcm'));`
 - `double` tvinger Matlab til at sætte variabel typen til `double`, som kan indeholde decimal-tal
 - `im2 = im/max(max(im));`
- `im2` vil nu indeholde pixels med værdier fra 0 til 1
- Dette kan plottes uden videre med `imshow`



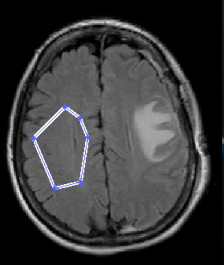
Masker og ROIs

- ROI = Region Of Interest
 - En markeret region, man ønsker at fx beregne middelværdi + standard deviation af
 - Kan tegnes via ROI funktioner
 - `impoly`
 - `imellipse`
 - `imrect`
 - `imfreehand`
 - etc.



Masker og ROIs

- `imshow(im)`
- `roi = impoly`
 - Klik på billedet punkt for punkt
 - Afslut ved at dobbeltklikke
- `roi` bliver nu til en variabel af typen `roi`




Masker og ROIs

- Vores `roi` kan konverteres til en maske
 - Billede med 1 og 0
 - 1 = indenfor `roi`
 - 0 = udenfor `roi`
 - `bw = createMask(roi)`
- Kan ganges på billedet
 - `m = bw .* im;`
 - `imshow(m)`



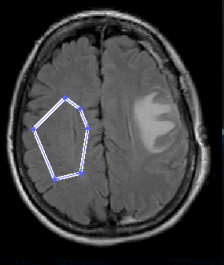
Beregning af middel og sd

- `idx = find(bw)`
 - Giver indekserne på de værdier i masken, som ikke er nul
- `im(idx)`
 - Giver gråtoneværdierne for de punkter, som er omsluttet af masken
- `mean(im(idx))`
 - Giver middelværdien for maskepunkterne
- `std(im(idx))`
 - Giver standard deviationen for maskepunkterne

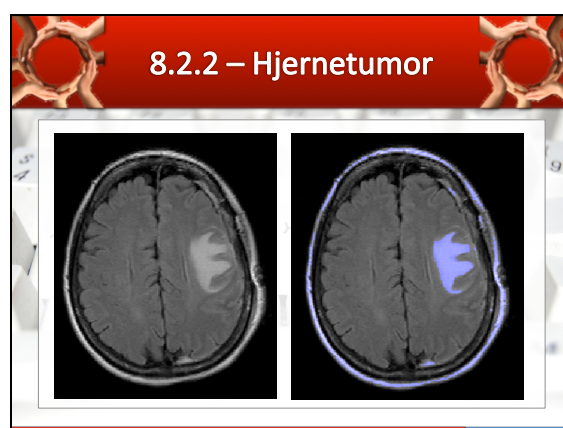


Øvelse: Hjernetumor

- Patienten har en svulst i højre side af billedet
- Kendetegnet ved forhøjet signalintensitet = lysere
- Marker normalt hjernevæv i venstre side med `roi`
- Beregn `m` og `sd` af `roi`
- Find pixels med værdi over `m + 2*sd`
- Marker dette som tumor
- Beregn arealet af denne




8.2.2 – Hjernetumor



Montage

- Et enkelt farve-billede:
 - Højde x bredde x 3
- Struktur til montage:
 - Højde x bredde x 3 x snit
- Montage-struktur for gråskala billeder
 - Højde x bredde x 1 x snit
- Kommando
 - `montage (imstack)`



Øvelse: Hjerte AAR

- Stak af MRI billeder, som dækker hjertet
- Væv i risiko for at dø = AAR = hvidt



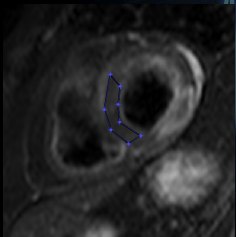
Myo Maske

- MyoMask = maske, som kun indeholder selve hjertevævet = myokardiet.
- `MyoMask .* im =` det segmenterede myokardium




Normal ROI

- Indtegn ROI med `impoly` af normalt myokardium:



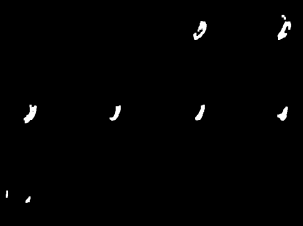
Enh

- Beregn Enhanced væv som det, der overstiger middel + 6*sd af det markerede ROI



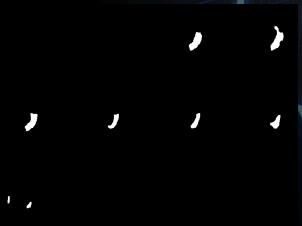
Brug af imopen

- For at fjerne støj, bruges `imopen`:



Brug af imclose

- For at lukke huller, bruges imclose



Dannelse af AARstak

- AARstak sættes til farvebillede, enhanced sættes til røde



8.2.3 – Hjerter AAR

