



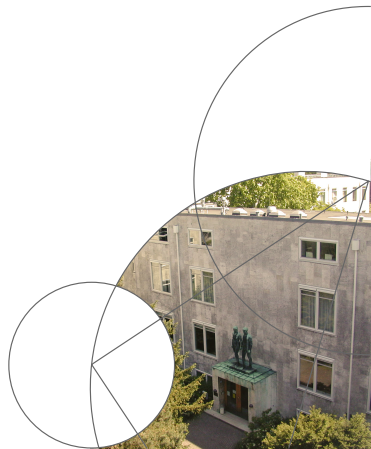
Det Naturvidenskabelige Fakultet



# Fagpakke dag 3

## Datanalyse med Machine Learning

Arinbjörn Brandsson  
Benjamin Rotendahl  
Mathias Mortensen  
Datalogisk Institut



- ① Hvad er Machine Learning
- ② Problemet
- ③ Algoritmen
- ④ Eksempel og algoritme-analyse



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.

Vi mennesker kan ikke overskue så store mængder af data



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.

Vi mennesker kan ikke overskue så store mængder af data

## ML til undsætning!

Vi ønsker istedet at lave systemer sådan at computere kan finde de underliggende mønstre og bruge den viden/erfaring der ligger i data'en.



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.

Vi mennesker kan ikke overskue så store mængder af data

## ML til undsætning!

Vi ønsker istedet at lave systemer sådan at computere kan finde de underliggende mønstre og bruge den viden/erfaring der ligger i data'en.

## Hvornår er ML godt?

- 1 Der eksisterer et mønster



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.

Vi mennesker kan ikke overskue så store mængder af data

## ML til undsætning!

Vi ønsker istedet at lave systemer sådan at computere kan finde de underliggende mønstre og bruge den viden/erfaring der ligger i data'en.

## Hvornår er ML godt?

- 1 Der eksisterer et mønster
- 2 Vi kan ikke finde en matematisk formel



# Hvad er Machine Learning

## Problemstilling

Vi indsamler større og større mængder af data hele tiden, så meget at det har fået sit eget buzzword **Big Data**.

Vi mennesker kan ikke overskue så store mængder af data

## ML til undsætning!

Vi ønsker istedet at lave systemer sådan at computere kan finde de underliggende mønstre og bruge den viden/erfaring der ligger i data'en.

## Hvornår er ML godt?

- 1 Der eksisterer et mønster
- 2 Vi kan ikke finde en matematisk formel
- 3 Vi har data på problemet





## Eksempel tid

*Netflix udlovede en dusør på 6,5 millioner kroner til den der kunne forbedre deres anbefalings algoritme med 10%.*



# Eksempel tid

*Netflix udlovede en dusør på 6,5 millioner kroner til den der kunne forbedre deres anbefalings algoritme med 10%.*

## Kan ML bruges?

- 1 Der eksisterer et mønster!



## Eksempel tid

*Netflix udlovede en dusør på 6,5 millioner kroner til den der kunne forbedre deres anbefalings algoritme med 10%.*

### Kan ML bruges?

- 1 Der eksisterer et mønster!
- 2 Vi kan ikke finde en formel for film



## Eksempel tid

*Netflix udlovede en dusør på 6,5 millioner kroner til den der kunne forbedre deres anbefalings algoritme med 10%.*

### Kan ML bruges?

- 1 Der eksisterer et mønster!
- 2 Vi kan ikke finde en formel for film
- 3 Der er massere af data til rådighed!



## Eksempel tid

*Netflix udlovede en dusør på 6,5 millioner kroner til den der kunne forbedre deres anbefalings algoritme med 10%.*

### Kan ML bruges?

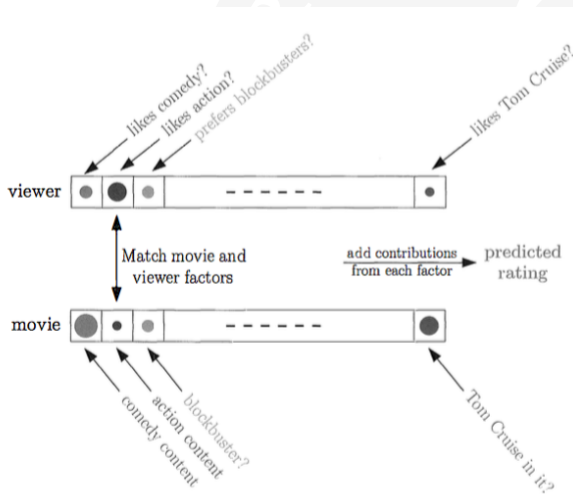
- 1 Der eksisterer et mønster!
- 2 Vi kan ikke finde en formel for film
- 3 Der er massere af data til rådighed!

*ML vandt konkurrencen!*



# Hvordan vandt de?

Figure : Netflix vinderen



# Dagens opgave

*Vi er blevet hyret af et hospital da de har hørt at vi dataloger kan hjælpe deres patienter.*



# Dagens opgave

*Vi er blevet hyret af et hospital da de har hørt at vi dataloger kan hjælpe deres patienter.*

## Problemstilling

Vi skal lave et system der, givet data om en patient, kan bestemme om deres svulst er godartet eller ondartet.





# Dagens opgave

*Vi er blevet hyret af et hospital da de har hørt at vi dataloger kan hjælpe deres patienter.*

## Problemstilling

Vi skal lave et system der, givet data om en patient, kan bestemme om deres svulst er godartet eller ondartet.

Hmm, det var da et ret generelt problem ...



# Dagens opgave

*Vi er blevet hyret af et hospital da de har hørt at vi dataloger kan hjælpe deres patienter.*

## Problemstilling

Vi skal lave et system der, givet data om en patient, kan bestemme om deres svulst er godartet eller ondartet.

Hmm, det var da et ret generelt problem ...

## Problemstilling

Vi skal lave et system der, givet data om en **kunde**, kan bestemme om det er en god forretning at låne dem penge.



# Dagens opgave

*Vi er blevet hyret af et hospital da de har hørt at vi dataloger kan hjælpe deres patienter.*

## Problemstilling

Vi skal lave et system der, givet data om en patient, kan bestemme om deres svulst er godartet eller ondartet.

Hmm, det var da et ret generelt problem ...

## Problemstilling

Vi skal lave et system der, givet data om en **kunde**, kan bestemme om det er en god forretning at låne dem penge.

Problemet hedder klassificering, gode løsninger kan have stor indflydelse inden for mange felter.



- ① Hvad er Machine Learning
- ② Problemet
- ③ Algoritmen
- ④ Eksempel og algoritme-analyse



# Håndtering af input og output

## Input

Threshold	1
Clump Thickness	7
Uniformity of Cell Size	1
Uniformity of Cell Shape	4
Epithelial Cell Size	2
Bare Nuclei	3
Bland Chromatin	8
Normal Nucleoli	10
Mitoses	3

## Output

Ondartet eller godartet 1



# Håndtering af input og output

## Input

Threshold	1
Clump Thickness	7
Uniformity of Cell Size	1
Uniformity of Cell Shape	4
Epithelial Cell Size	2
Bare Nuclei	3
Bland Chromatin	8
Normal Nucleoli	10
Mitoses	3



## Data vektor

$$\begin{pmatrix} 1 \\ 7 \\ 1 \\ 4 \\ 2 \\ 3 \\ 8 \\ 10 \\ 3 \end{pmatrix}$$

## Output

Ondartet eller godartet 1



## Output

1

# Formalisering

## Termer

**Input** En vektor (patient data)



# Formalisering

## Termer

**Input** En vektor (patient data)

**Output** 1 eller  $-1$  (ondartet eller godartet)





# Formalisering

## Termer

**Input** En vektor (patient data)

**Output** 1 eller  $-1$  (ondartet eller godartet)

**Læringsmål**  $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$



# Formalisering

## Termer

**Input** En vektor (patient data)

**Output** 1 eller  $-1$  (ondartet eller godartet)

**Læringsmål**  $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$

**Data**  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  (Hvad vi lærer fra)



# Formalisering

## Termer

**Input** En vektor (patient data)

**Output** 1 eller  $-1$  (ondartet eller godartet)

**Læringsmål**  $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$

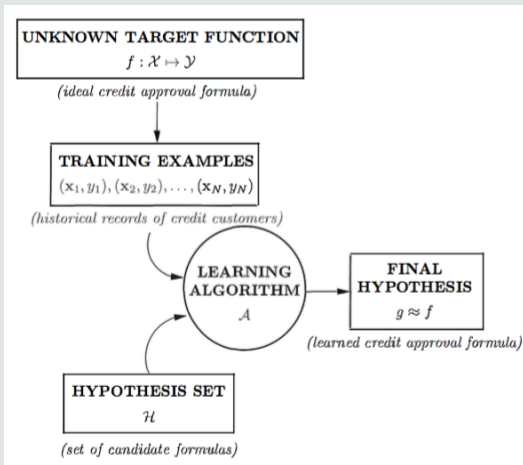
**Data**  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  (Hvad vi lærer fra)

**Hypotese**  $g : \mathcal{X} \rightarrow \mathcal{Y}$  (Vores systems "Hjerne")



# Visual Formalisering

Figure : Visuelt læringsdiagram



- 1 Hvad er Machine Learning
- 2 Problemet
- 3 Algoritmen**
- 4 Eksempel og algoritme-analyse



# Valget af lærings-algoritmen

## Perceptron

Den laver et *hyperplan* der adskiller data'en og finder en opdeling der giver en **lav fejl**.



# Valget af lærings-algoritmen

## Perceptron

Den laver et *hyperplan* der adskiller data'en og finder en opdeling der giver en **lav fejl**.

Tænk på den som en form for lineær regression på steroider

$$y = ax + b$$



# Valget af lærings-algoritmen

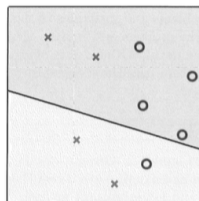
## Perceptron

Den laver et *hyperplan* der adskiller data'en og finder en opdeling der giver en **lav fejl**.

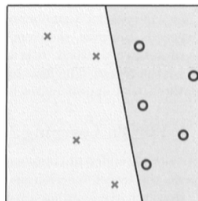
Tænk på den som en form for lineær regression på steroider

$$y = ax + b$$

## Eksempel på algoritmen



(a) Misclassified data



(b) Perfectly classified data





# Algoritmen i ord

## Hvordan virker den?

Vi har en masse vektorer  $v_1, v_2, \dots, v_n$  og en liste af svar  $y_1, y_2, \dots, y_n$ .



# Algoritmen i ord

## Hvordan virker den?

Vi har en masse vektorer  $v_1, v_2, \dots, v_n$  og en liste af svar  $y_1, y_2, \dots, y_n$ .

Vi lader  $w$  være vores “vægt-vektor”.



# Algoritmen i ord

## Hvordan virker den?

Vi har en masse vektorer  $v_1, v_2, \dots, v_n$  og en liste af svar  $y_1, y_2, \dots, y_n$ .

Vi lader  $w$  være vores “vægt-vektor”.

$$\text{Godartet svulst : } \sum_{i=1}^d w_i x_i > b$$

$$\text{Ondartet svulst : } \sum_{i=1}^d w_i x_i < b$$



# Algoritmen i ord

## Hvordan virker den?

Vi har en masse vektorer  $v_1, v_2, \dots, v_n$  og en liste af svar  $y_1, y_2, \dots, y_n$ .

Vi lader  $w$  være vores “vægt-vektor”.

$$\text{Godartet svulst : } \sum_{i=1}^d w_i x_i > b$$

$$\text{Ondartet svulst : } \sum_{i=1}^d w_i x_i < b$$

Vores hypotese bliver så

$$h(x) = \text{fortegn} \left( \sum_{i=0}^d w_i x_i \right)$$



# Algoritmen i ord

## Hvordan virker den?

Vi har en masse vektorer  $v_1, v_2, \dots, v_n$  og en liste af svar  $y_1, y_2, \dots, y_n$ .

Vi lader  $w$  være vores “vægt-vektor”.

$$\text{Godartet svulst : } \sum_{i=1}^d w_i x_i > b$$

$$\text{Ondartet svulst : } \sum_{i=1}^d w_i x_i < b$$

Vores hypotese bliver så

$$h(x) = \text{fortegn} \left( \sum_{i=0}^d w_i x_i \right)$$

Men hvordan bestemmer vi  $w$ ?



# Hvordan den lærer

## Hvordan $w$ bestemmes



# Hvordan den lærer

## Hvordan $w$ bestemmes

$w =$  vælg tilfældige tal



# Hvordan den lærer

## Hvordan $w$ bestemmes

$w =$  vælg tilfældige tal

Vi forbedrer  $w$  hver gang!





# Hvordan den lærer

## Hvordan $w$ bestemmes

$w =$  vælg tilfældige tal

Vi forbedrer  $w$  hver gang!

Hvis  $x'$  er på den forkerte side af  $w$  så lærer den  
“erfaringen” ved formlen



# Hvordan den lærer

## Hvordan $w$ bestemmes

$w =$  vælg tilfældige tal

Vi forbedrer  $w$  hver gang!

Hvis  $x'$  er på den forkerte side af  $w$  så lærer den “erfaringen” ved formlen

$$w_{ny} = w + y'x'$$



# Hvordan den lærer

## Hvordan $w$ bestemmes

$w =$  vælg tilfældige tal

Vi forbedrer  $w$  hver gang!

Hvis  $x'$  er på den forkerte side af  $w$  så lærer den “erfaringen” ved formlen

$$w_{ny} = w + y'x'$$

Forsæt med at lære indtil du ikke kan lære mere.



# Perceptron algoritme

## Pseudocode

### Algorithm 1

Input: datasæt  $X = [(x_1, y_1), \dots, (x_n, y_n)]$

Output: Hypotesen  $w$ .

$w$  = Tilfældige tal

$\text{misCat} = (1, 1)$

**while**  $\text{misCat} \neq (0, 0)$  **do**

$\text{misCat} = (0, 0)$

**for**  $(x_i, y_i)$  in  $X$  **do**

**if**  $\text{sign}(w^T x_i) \neq y_i$  **then**

$\text{misCat} = (x_i, y_i)$

$w = w + y_i x_i$

**end if**

**end for**

**end while**

**return**  $w$

- ① Hvad er Machine Learning
- ② Problemet
- ③ Algoritmen
- ④ Eksempel og algoritme-analyse



# Vi prøver at køre den!

*Eksempel i MatLab*



# Vi prøver at køre den!

*Eksempel i MatLab*

## Analyse

Nogen der kan gætte køretiden?



# Vi prøver at køre den!

*Eksempel i MatLab*

## Analyse

Nogen der kan gætte køretiden?

$$O(2^{(n+1)\log(n+1)}(n+1)^2)$$





# Vi prøver at køre den!

*Eksempel i MatLab*

## Analyse

Nogen der kan gætte køretiden?

$$O(2^{(n+1)\log(n+1)}(n+1)^2)$$

Redder vi så nogle liv?



# Vi prøver at køre den!

*Eksempel i MatLab*

## Analyse

Nogen der kan gætte køretiden?

$$O(2^{(n+1)\log(n+1)}(n+1)^2)$$

Redder vi så nogle liv? Lad os kode det og se hvor god den er!



# Afslutning

## Hvor god er den?

I opgaverne kigger i kun på **25 eksempler!** og tester på 75 patienter



# Afslutning

## Hvor god er den?

I opgaverne kigger i kun på **25 eksempler!** og tester på 75 patienter

I kan forvente at den har ret på cirka 60 – 70% af patienterne!.



# Afslutning

## Hvor god er den?

I opgaverne kigger i kun på **25 eksempler!** og tester på 75 patienter

I kan forvente at den har ret på cirka 60 – 70% af patienterne!.

Kører man den istedet med 500 eksempler og tester på 180.



# Afslutning

## Hvor god er den?

I opgaverne kigger i kun på **25 eksempler!** og tester på 75 patienter

I kan forvente at den har ret på cirka 60 – 70% af patienterne!.

Kører man den istedet med 500 eksempler og tester på 180. Rammer den rigtigt 181 gange og forkert 2 gange.

Det betyder at den har en succes rate på **98,9%!**



# Afslutning

## Hvor god er den?

I opgaverne kigger i kun på **25 eksempler!** og tester på 75 patienter

I kan forvente at den har ret på cirka 60 – 70% af patienterne!.

Kører man den istedet med 500 eksempler og tester på 180. Rammer den rigtigt 181 gange og forkert 2 gange. Det betyder at den har en succes rate på **98,9%!**

Spørgsmål?



# Evaluering

## Facebook grupper

<https://www.facebook.com/DIKUDatalogi/>

<https://www.facebook.com/groups/Datalogi.I.Gymnasiet>





# Evaluering

## Facebook grupper

<https://www.facebook.com/DIKUDatalogi/>

<https://www.facebook.com/groups/Datalogi.I.Gymnasiet>

## Evaluerings skema

<http://rotendahl.dk/eval>



# Evaluering

## Facebook grupper

<https://www.facebook.com/DIKUDatalogi/>

<https://www.facebook.com/groups/Datalogi.I.Gymnasiet>

## Evaluerings skema

<http://rotendahl.dk/eval>

Tak for denne gang! (Vi ses på DIKU)

