Data Advanced

Hoofdstuk 3

Machine Learning

DE HOGESCHOOL
MET HET NETWERK

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook

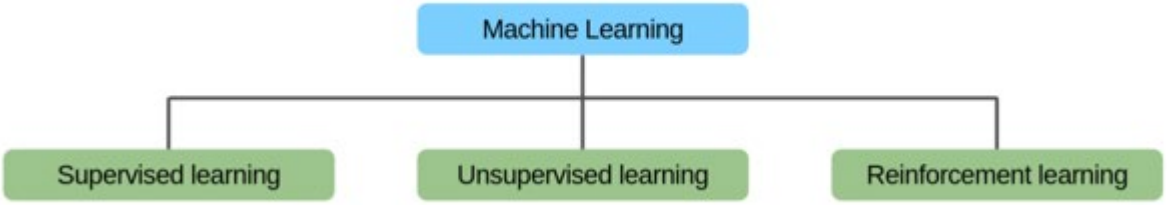
Inhoud

1. Inleiding
2. Historiek
3. ML problemen herkennen
4. Het ML proces
    1. Type ML - problemen
    2. Data
    3. ML - algoritme
5. Types van ML – problemen
    1. Supervised: Classificatie
    2. Supervised: Regressie
    3. Unsupervised: Clustering
    4. Recommendations

1. Inleiding (pg 117)

- Frequent gehoorde term
- Machines zelfstandig taken uitvoeren

  leert van informatie en zoekt patronen
- Wiskunde – statistiek?
- Soorten:



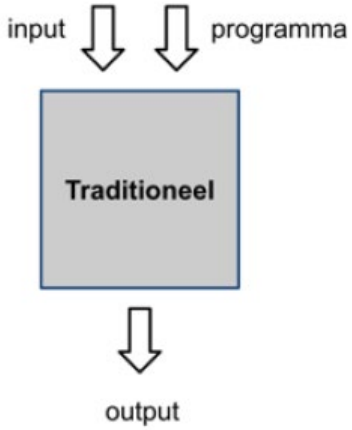
2. Historiek (pg 118)

- Arthur Samuel (jaren ‘50): schaakspel
- Frank Rosenblatt: perceptron (1958) = 1ste kunstmatige neurale netwerk
- Stanford University (1979): bewegende robot

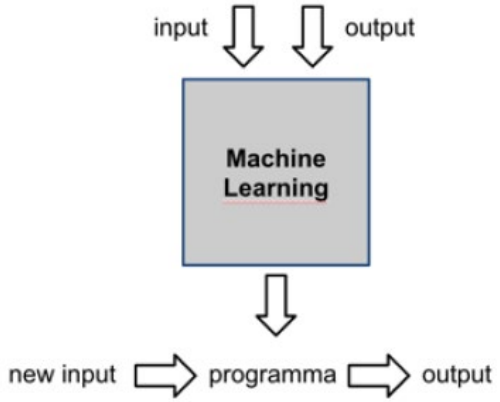
3. ML problemen herkennen(pg 120)

- Voorbeeld: Alien

Klassieke aanpak



ML aanpak



Defintie: ML is het proces waarbij een computerprogramma of systeem in staat is te leren hoe een taak uit te voeren en dit door ervaring. Ervaring voor een computer = DATA

3. ML problemen herkennen(pg 123)

- **Rule based**: je past een aantal statische regels toe op de huidige context (tijdstip, dag, …).

- **ML**: Ook in de ML benadering pas je een aantal regels toe maar het verschil is dat deze regels (automatisch) geüpdatet worden op basis van nieuwe data.
- 3 stappen:
  - verzamel een grote set data
  - Gebruik een algoritme dat “zelfstandig” een verband vindt
  - Update dit verband voortdurend mbv nieuwe data

3. ML problemen herkennen (pg 124)

- Netflix
- E-mail: spam ⇔ geen spam
- Zelfrijdende auto
- Slimme thermostaat
- Gezichtsherkenning

3. ML problemen herkennen (pg 126)

Vuistregels gebruik ML:

- het is moeilijk om het probleem te gieten in regels
- je beschikt over een grote set historische data
- de patronen of relaties tussen de data zijn dynamisch

4. Het ML proces (pg 127 - 129)

- Type

  Supervised: input EN output

  - inputvariabelen $(x_1, x_2, \dots, x_n)$ EN outcome variabele *y* is aanwezig in de training data set
  - functie *f* die het verband geeft nl $y = f(x_1, x_2, \dots, x_n)$
  - functie *f* voorspelt op basis van een nieuwe income $(x_1, x_2, \dots, x_n)$ de outcome *y*
  - de functie *f* wordt continu bijgestuurd

4. Het ML proces (pg 127 - 129)

- Type

  Unsupervised: enkel input

  - Enkel de inputvariabele $(x_1, x_2, \ldots, x_n)$ is aanwezig in de training data set (geen outcome *y*)
  - Het algoritme brengt patronen, structuren, groepen, … naar voren die op voorhand niet opgenomen zijn in de dataset.

- Data

- Algoritme

5. ML problemen uitgewerkt(pg 130)

- 5.1 Supervised ML: Classificatie
- 5.2 Supervised ML: Regressie
- 5.3 Unsupervised ML: Clustering
- 5.4 Recommendations

5.1 Classificatie (pg 131)

- Doel

- Soorten:
  - binary classificatie
  - multi-class classificatie
  - multi-label classificatie
  - multi-output classificatie

5.1 Classificatie (pg 131)

- Classifier
- Features
- Training data
- Testing data

5.1. Classificatie (pg 133)

6 verschillende classificatie algoritmes:

5.1.1 Naive Bayes algoritme

5.1.2 Support Vector Machine algoritme

5.1.3 Decision Tree algoritme

5.1.4 Logistic Regression algoritme

5.1.5 Linear Discriminant Analysis

5.1.6 Nearest Neighbour algoritme

5.1.1 Naive Bayes (pg 134)

- Basis: Voorwaardelijke kans + regel van Bayes
- Vb: Persoon classificatie (pg 134)
- Vb: Sentiment Analysis (pg 136)

Naive Bayes – Persoon classificatie

Politieagent – jogger?

Attribuut	Politieagent	jogger
Handboeien	6	0
sportschoenen	2	8
Wapen	9	0
badge	8	3
Walkietalkie	8	0

Stap 1: bereken $P(jogger \mid handboeien, badge)$ via de wet van Bayes

Stap 2: bereken $P(politieagent \mid handboeien, badge)$ via de wet van Bayes

Stap 3: vergelijk deze kansen

Naive Bayes – Sentiment Analysis

- Term frequency representation
  - Lijst met ALLE woorden uit training data:

(hallo, dit, zijn, alle, woorden, die, kunnen, voorkomen, in, een, tekst, en, ook, is)

  - Te analyseren tekst:

(hallo, dit, is, een, tekst, en, een, tekst) → *(1, 1, 0, 0, 0, 0, 0, 0, 0, 2, 2, 1, 0, 1)*

- Trainingsfase

Income	Outcome
tekst 1 $\xrightarrow{term\ freq\ repr}$ (1,0,3,1,…,0)	Positief
tekst 2 $\xrightarrow{term\ freq\ repr}$ (1,1,2,1,…,1)	Positief
tekst 3 $\xrightarrow{term\ freq\ repr}$ (2,0,1,0,…,0)	Negatief
…	
tekst 1000 $\xrightarrow{term\ freq\ repr}$ (1,0,2,2,…,3)	Positief

Naive Bayes – Sentiment Analysis

- Positiviteitskans elk woord via wet van Bayes

$$P_{blij} = P(blij|Pos\ tekst)$$

$$= \frac{Som\ van\ alle\ woordfreq\ van\ blij\ in\ pos\ teksten}{Som\ van\ alle\ woordfreq\ van\ blij\ in\ hele\ training\ data}$$

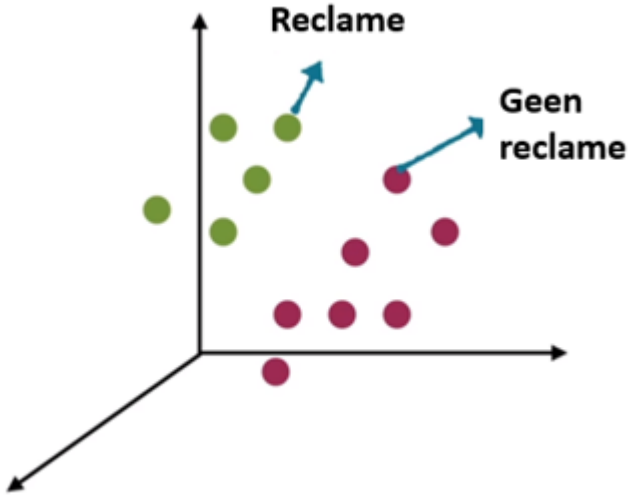
	Positiviteitskans
blij	0.92
geweldig	0.95
pxl	0.81
data	0.99
saai	0.1
moeilijk	0.33
…	

- Testfase: vb: data is geweldig
  - ✓ $PosScore = P_{data} * P_{geweldig} * P_O = 0.99 * 0.95 * 0.55 = 0.51$
  - ✓ $NegScore = (1 - P_{data}) * (1 - P_{geweldig}) * (1 - P_O) = 0.01 * 0.05 * 0.45$

    $= 0.000225$
  - ✓ Vergelijking PosScore - NegScore

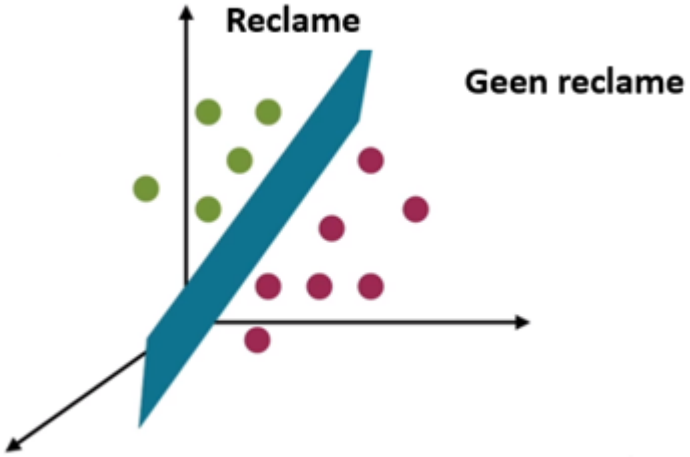
5.1.2 Support Vector Machine (pg 141)

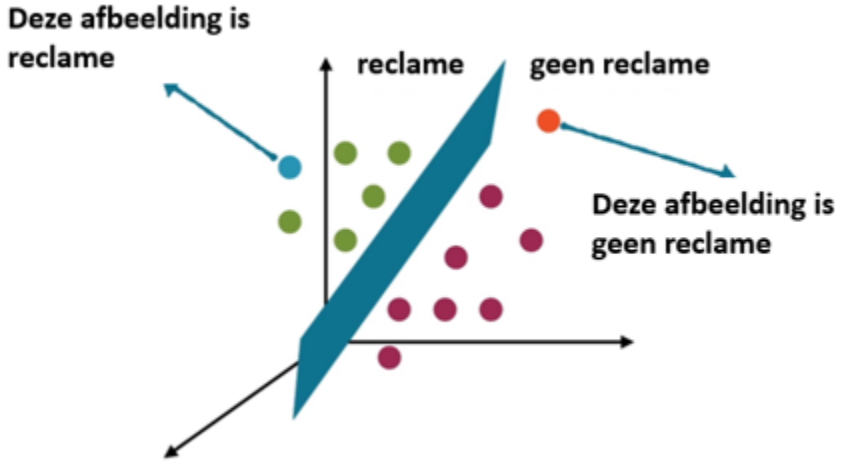
- Basis: grensvlak
- Vb: reclame detectie



5.1.2 Support Vector Machine (pg 141)

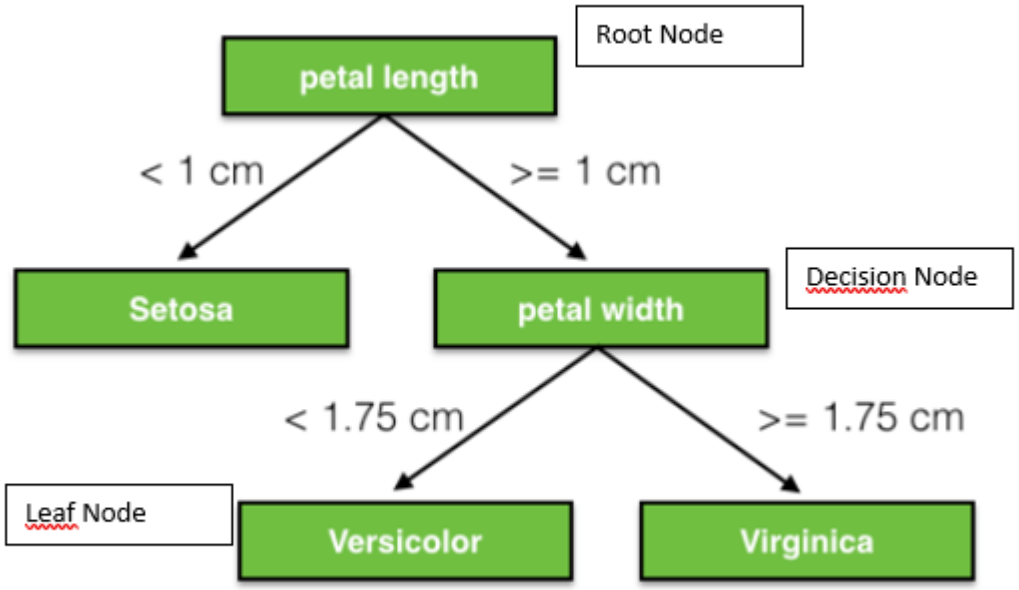
- Vb: reclame detectie





5.1.3 Decision Tree (pg 143)

- Basis: Boomstructuur



- Jupyter notebook: Machine_Learning_1_Classification_Iris

5.1.4 Logistic Regression (pg 144)

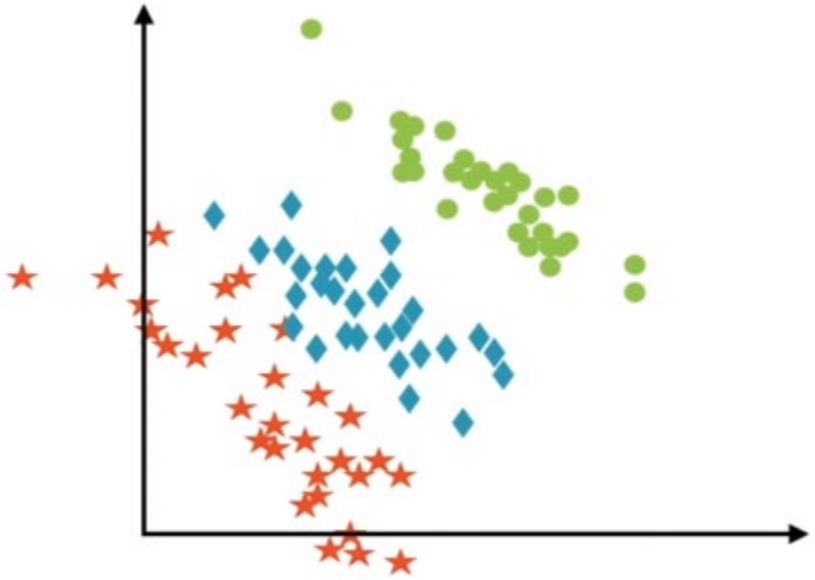
- Basis: Logistic function $$y = \frac{e^{b_0 + b_1 x}}{1 + e^{b_0 + b_1 x}}$$

- Binary classification

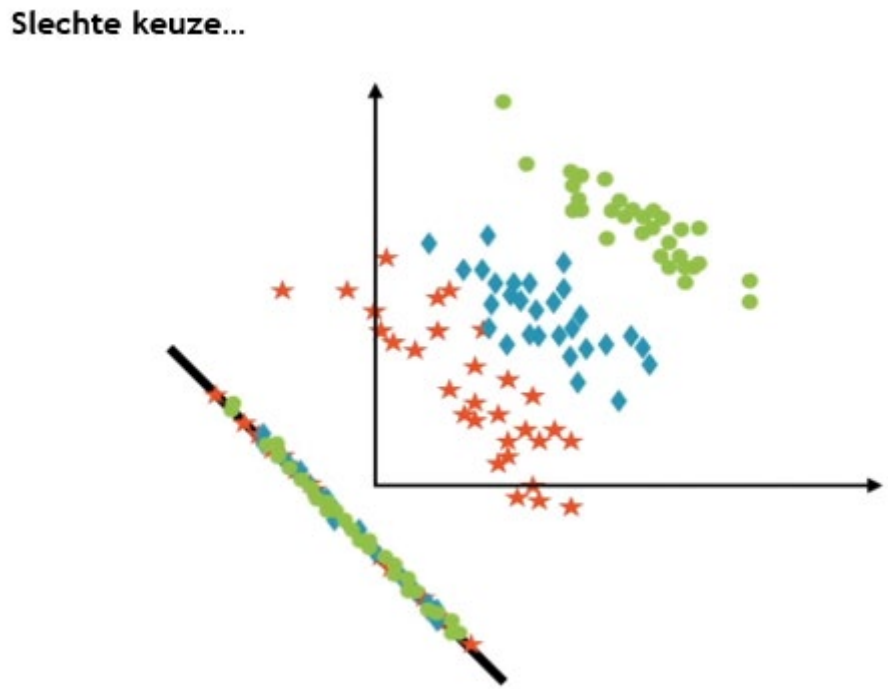
- Jupyter notebook: Machine_Learning_2_Classification_Titanic

5.1.5 Linear Discriminant (pg 145)

- Basis: maximalisatie van afstanden

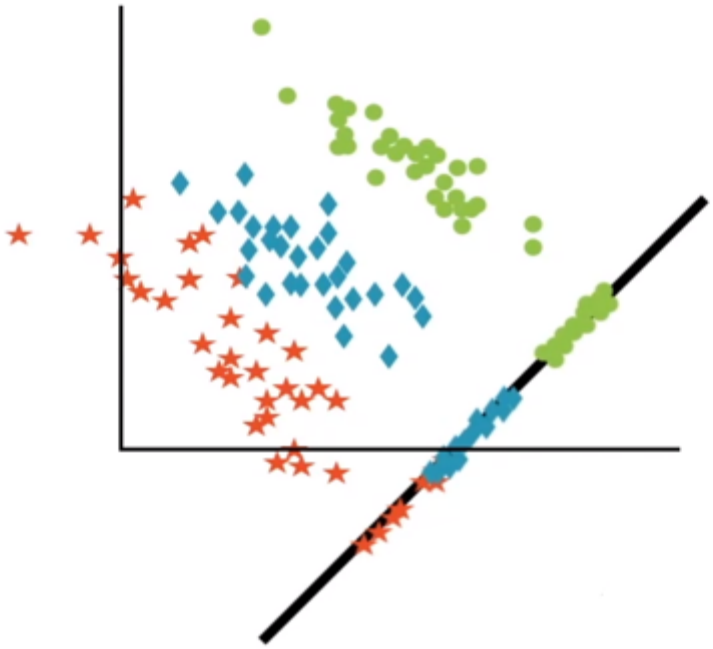


5.1.5 Linear Discriminant (pg 145)



5.1.5 Linear Discriminant (pg 145)

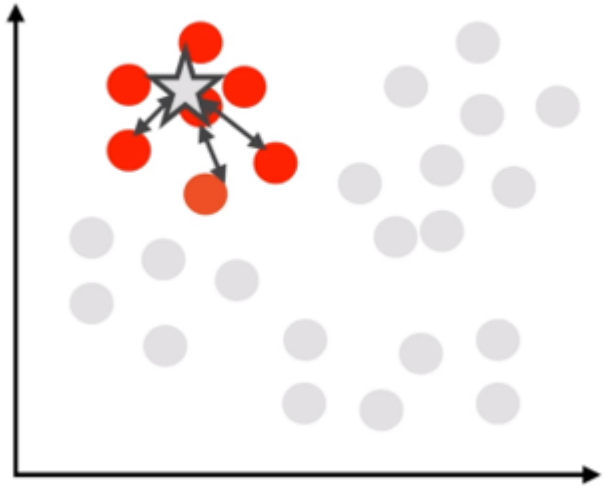
Beste keuze...



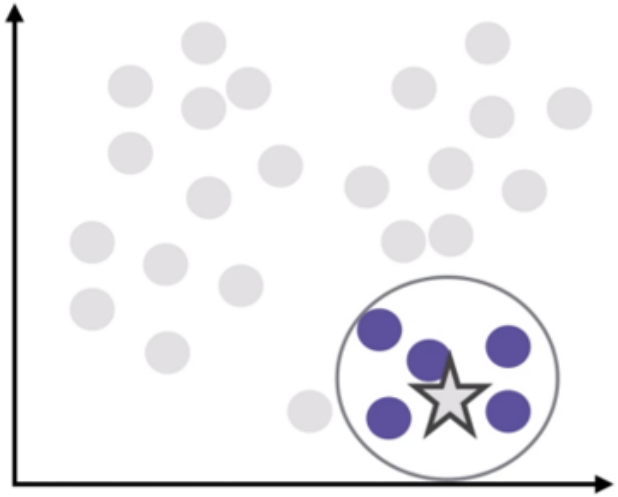
5.1.6 Nearest Neighbor (pg 147)

- Basis: Minimalisatie van afstand

K - nearest neighbor classificatie

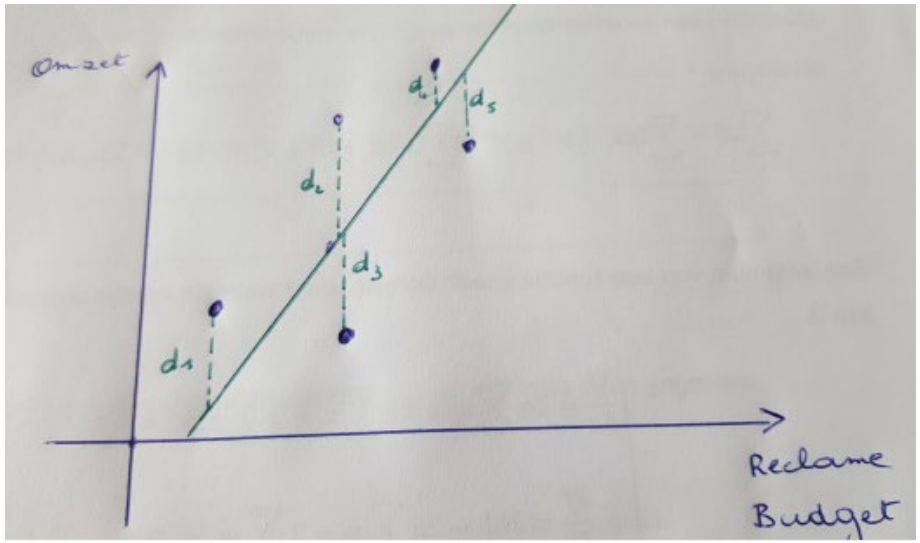


Radius nearest neighbor classificatie



5.2 Regressie (pg 148)

- Supervised ML: input + output in dataset
- Doel: continue variabele voorspellen
- Eenvoudige lineaire regressie mbv de methode van de kleinste kwadraten



- Jupyter notebook: Machine_Learning_3_Regression_Boston

5.3 Clustering (pg 153)

- Unsupervised ML
- Doel: items groeperen (clusters = niet gekend)
- Basisidee
- Globale werkwijze

5.3 Clustering (pg 153)

- Globale werkwijze

5 clusters

2 clusters

5.3 Clustering (pg 155)

- Bestaande clustering algoritmes
  - K – means clustering
  - K – median clustering
  - Hiërarchical clustering
  - Density-based clustering
  - Distribution-based clustering

5.3 Clustering (pg 156)

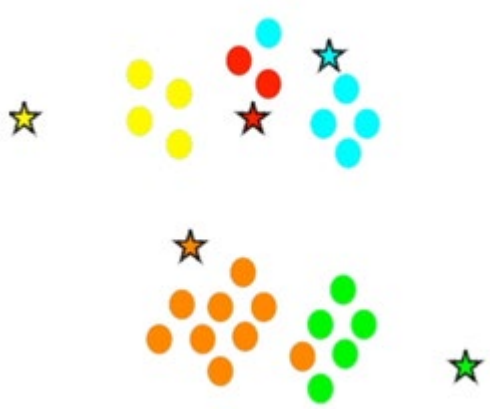
K – Means algoritme voor het clusteren van documenten

- Dataset
- Features
- Algoritme
  - Stap 1

5.3 Clustering (pg 156)

K – Means algoritme voor het clusteren van documenten

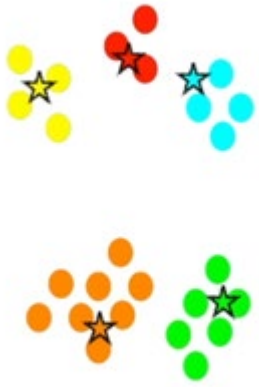
- Algoritme
  - Stap 2
  - Stap 3



5.3 Clustering (pg 156)

K – Means algoritme voor het clusteren van documenten

- Algoritme
  - Convergentie



5.4 Recommendations (pg 158)

- Doel:
  - Aanbevelingen doen
  - Klanten trouw laten zijn aan

- Voorbeeld
  - Netflix – Spotify
  - Dynamische websites

5.4 Recommendations (pg 158)

- Hoe vinden we de top 10 films die interessant zijn voor een gebruiker?
- Definitie Collaborative Filtering
- Basisaanname Collaborative Filtering

Jupyter Notebook: Machine_Learning_4_Recommendations_Movie