

Bewegingen herkennen met een smartphone

Arne De Brabandere
Menno Keustermans

Begeleiders:
Wannes Meert
Leander Schietgat

Doelstellingen

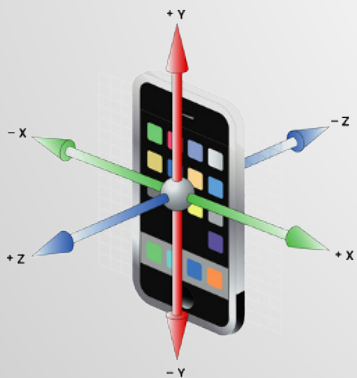
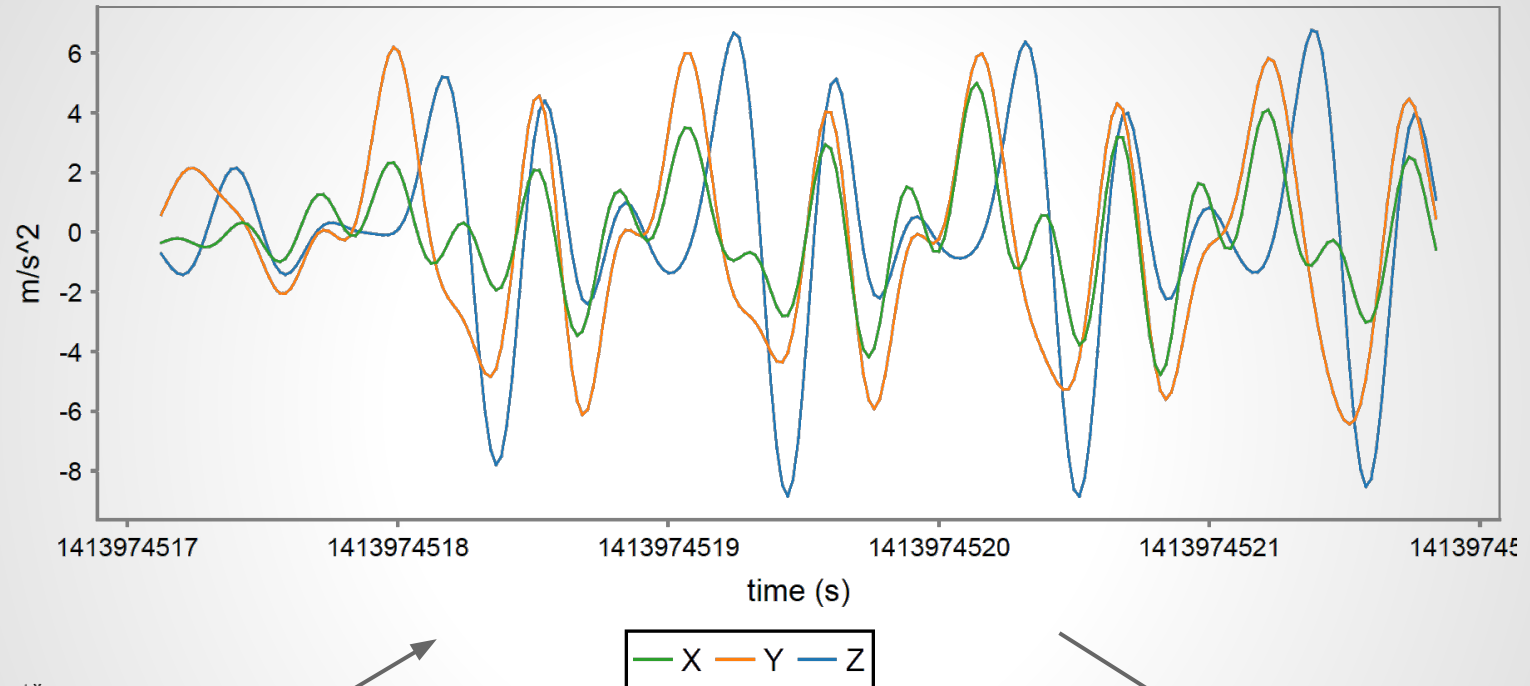
Welke activiteiten kunnen herkend worden?

- wandelen - lopen
- trap op - trap af
- fietsen
- ...

Kan een sequentie van bewegingen geclassificeerd worden?

+ Hoe nauwkeurig?

Probleemstelling

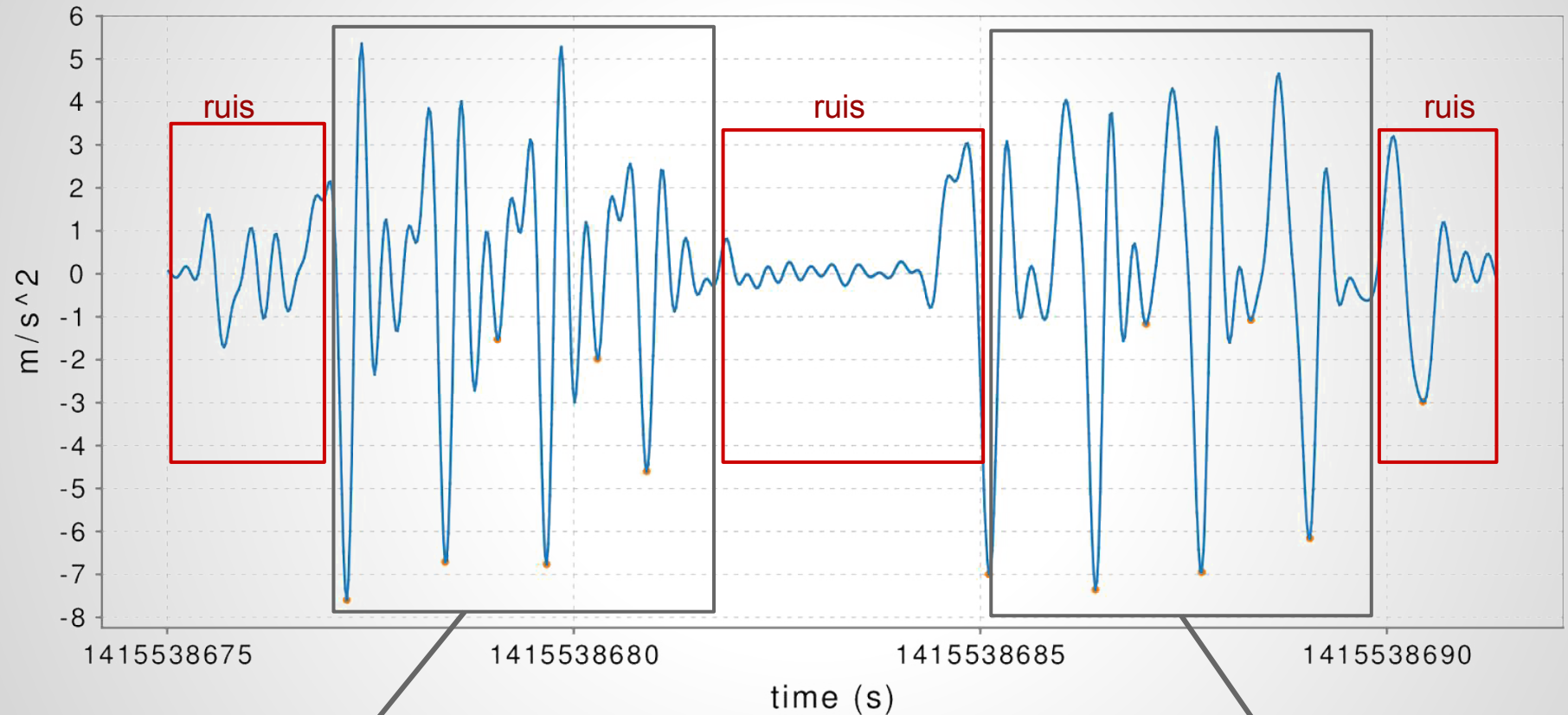


“wandelen”

Herkennen van de beweging

Probleemstelling: sequentie

z



“wandelen”

“trap op”

Inhoud

- Papers
- Gegevens opmeten
- Features
- Machine Learning technieken

Papers

Features

+

ML-technieken

- **Activity Recognition from Accelerometer Data -**
Nishkam Ravi and Nikhil Dandekar and Preetham Mysore and Michael L. Littman
- **Feature Processing and Modeling for 6D Motion Gesture Recognition -**
Mingyu Chen, Ghassan AlRegib en Biing-Hwang Juang
- **A Hybrid Discriminative & Generative Approach for Modeling Human Activities -**
Jonathan Lester, Tanzeem Choudhury, Nicky Kern, Gaetano Borriello en Blake Hannaford
- **An introduction to inertial navigation -**
Oliver J. Woodman

Papers

- **Activity Recognition from Accelerometer Data -**
Nishkam Ravi and Nikhil Dandekar and Preetham Mysore and Michael L. Littman

- **Feature Processing and Modeling for 6D Motion Gesture Recognition -**

Mingyu Chen, Ghassan AlRegib en Biing-Hwang Juang

- **A Hybrid Discriminative & Generative Approach for Modeling Human Activities -**

Jonathan Lester, Tanzeem Choudhury, Nicky Kern, Gaetano Borriello en Blake Hannaford

- **An introduction to inertial navigation -**

Oliver J. Woodman

HMM's

Papers

- **Activity Recognition from Accelerometer Data -**
Nishkam Ravi and Nikhil Dandekar and Preetham Mysore and Michael L. Littman
- **Feature Processing and Modeling for 6D Motion Gesture Recognition -**
Mingyu Chen, Ghassan AlRegib en Biing-Hwang Juang
- **A Hybrid Discriminative & Generative Approach for Modeling Human Activities -**
Jonathan Lester, Tanzeem Choudhury, Nicky Kern, Gaetano Borriello en Blake Hannaford
- **An introduction to inertial navigation -**
Oliver J. Woodman

Positie
bepalen

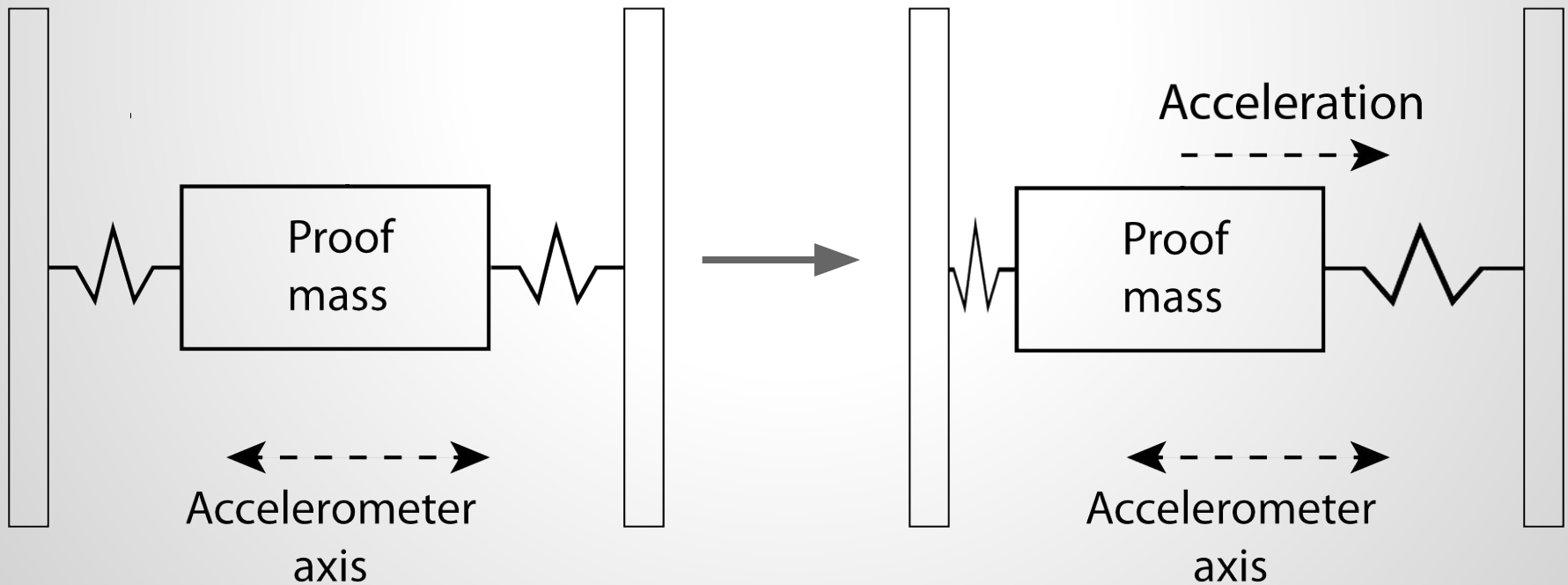


Gegevens opmeten

Bewegingen detecteren en normaliseren

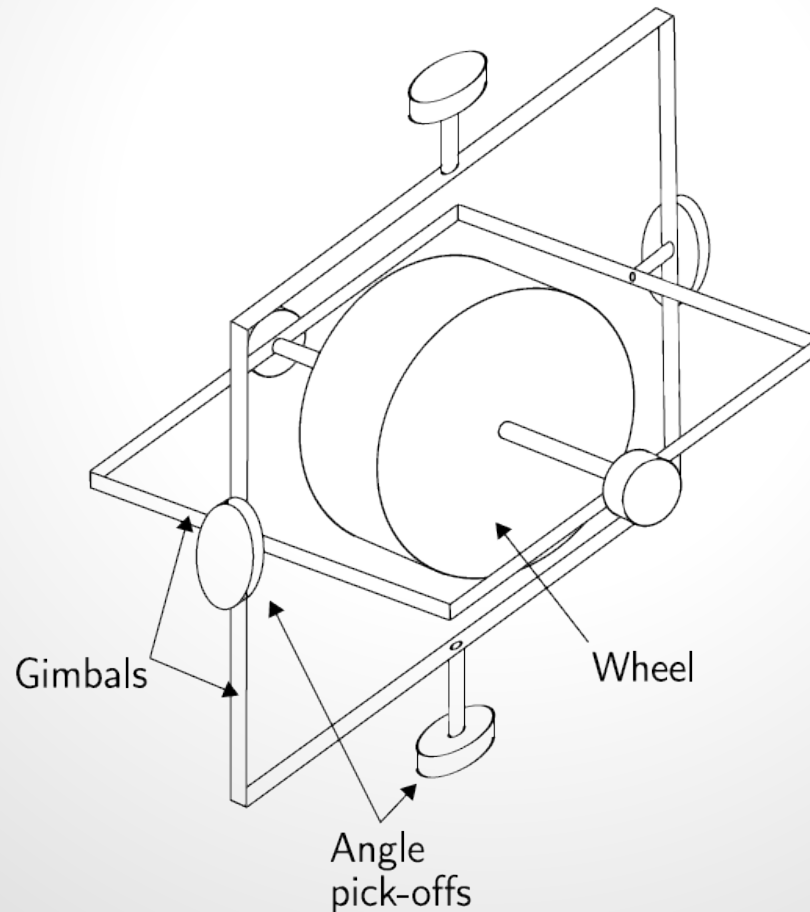
Accelerometer

Detecteert versnelling in 3D (x, y, z)



Gyroscoop

Meet de hoeksnelheid in 3D (roll, pitch, yaw)



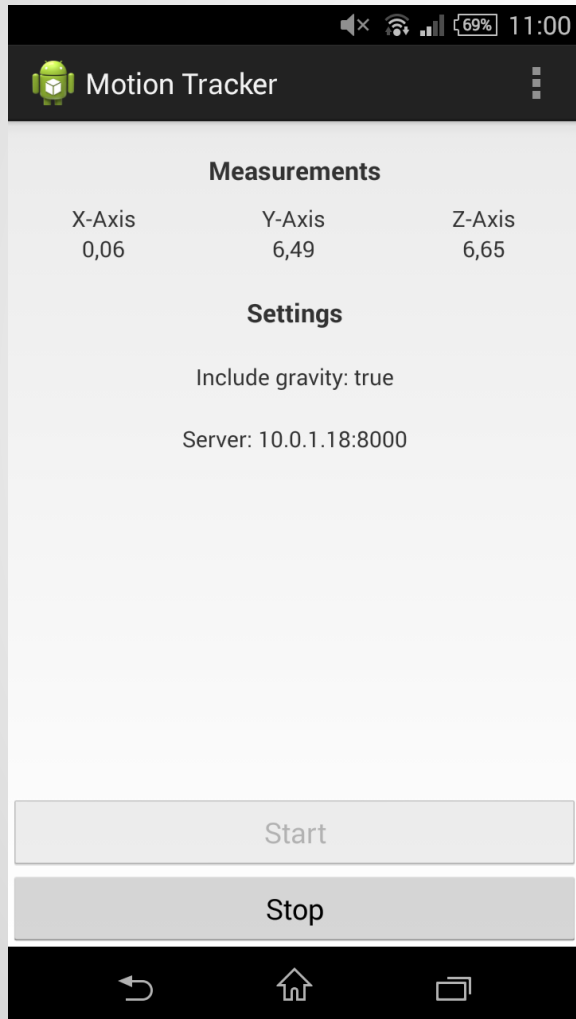
6D Bewegingsdetector: sensor fusion

Combinatie van gegevens van accelerometer en gyroscoop

Gegevens normaliseren:

beweging detecteren ongeacht de positie van de smartphone in de broekzak

Applicatie

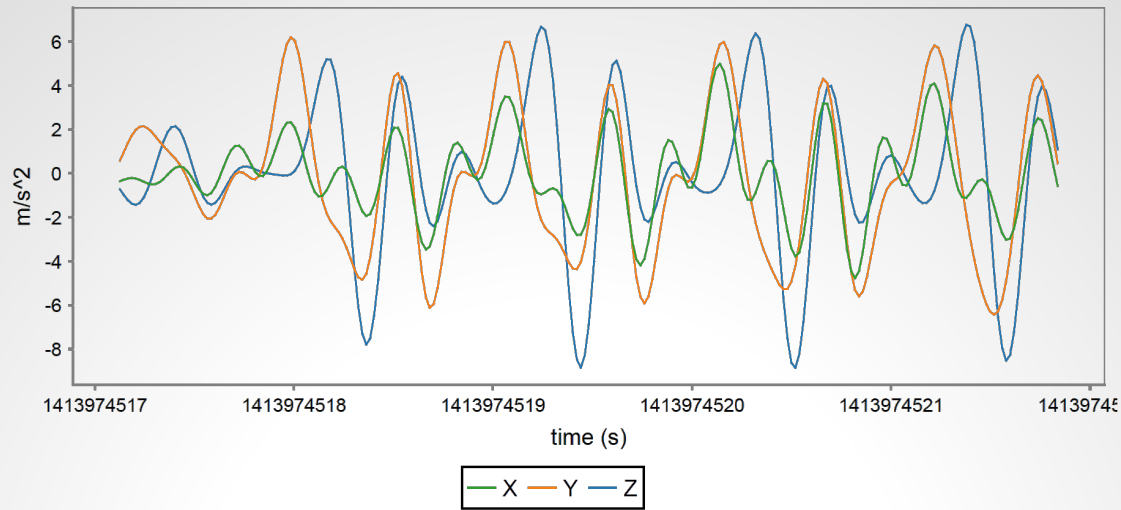


Meet de versnelling en de rotatie van de smartphone aan 50 hz

= standaard sampling frequentie

Typische output:

```
{"timestamp": ... ,  
"q": [..., ..., ..., ...],  
"x": ..., "y": ..., "z": ...}
```



Features

Parameters berekenen uit
accelerometerdata

Berekend met behulp van tool:
MotionFingerprint (door Wannes Meert)

Statistische variabelen

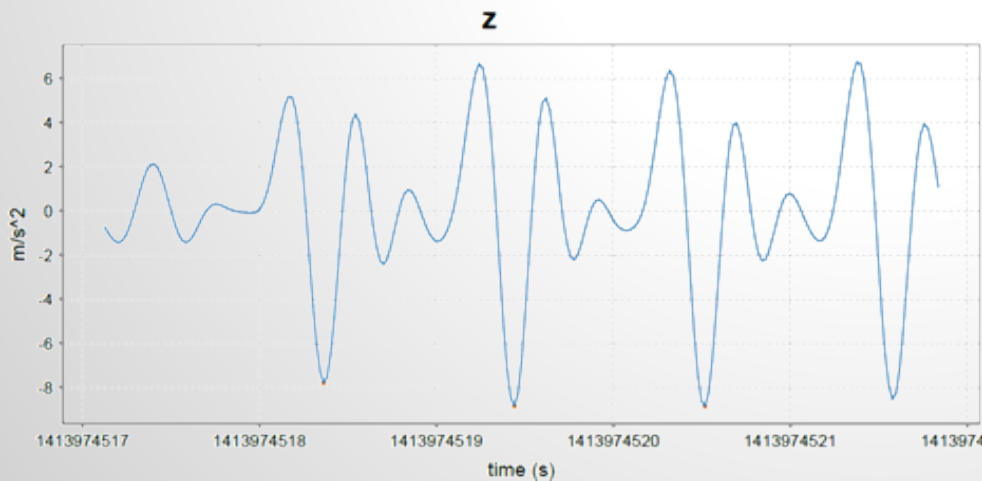
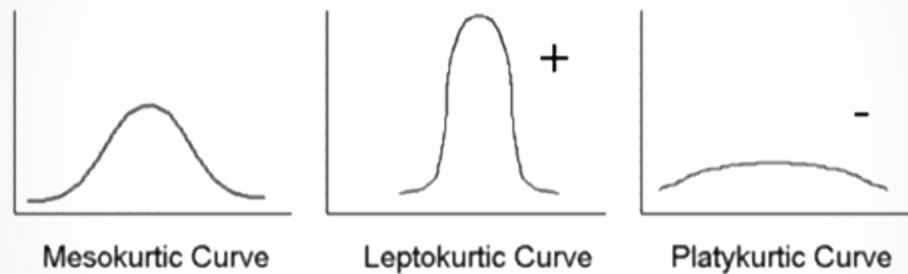
- Gemiddelde
- Standaardafwijking
- Correlatie

Berekend voor de versnelling:

$\left\{ \begin{array}{l} \text{in de z-richting} \\ \text{in het xy-vlak} \end{array} \right.$

Kurtosis

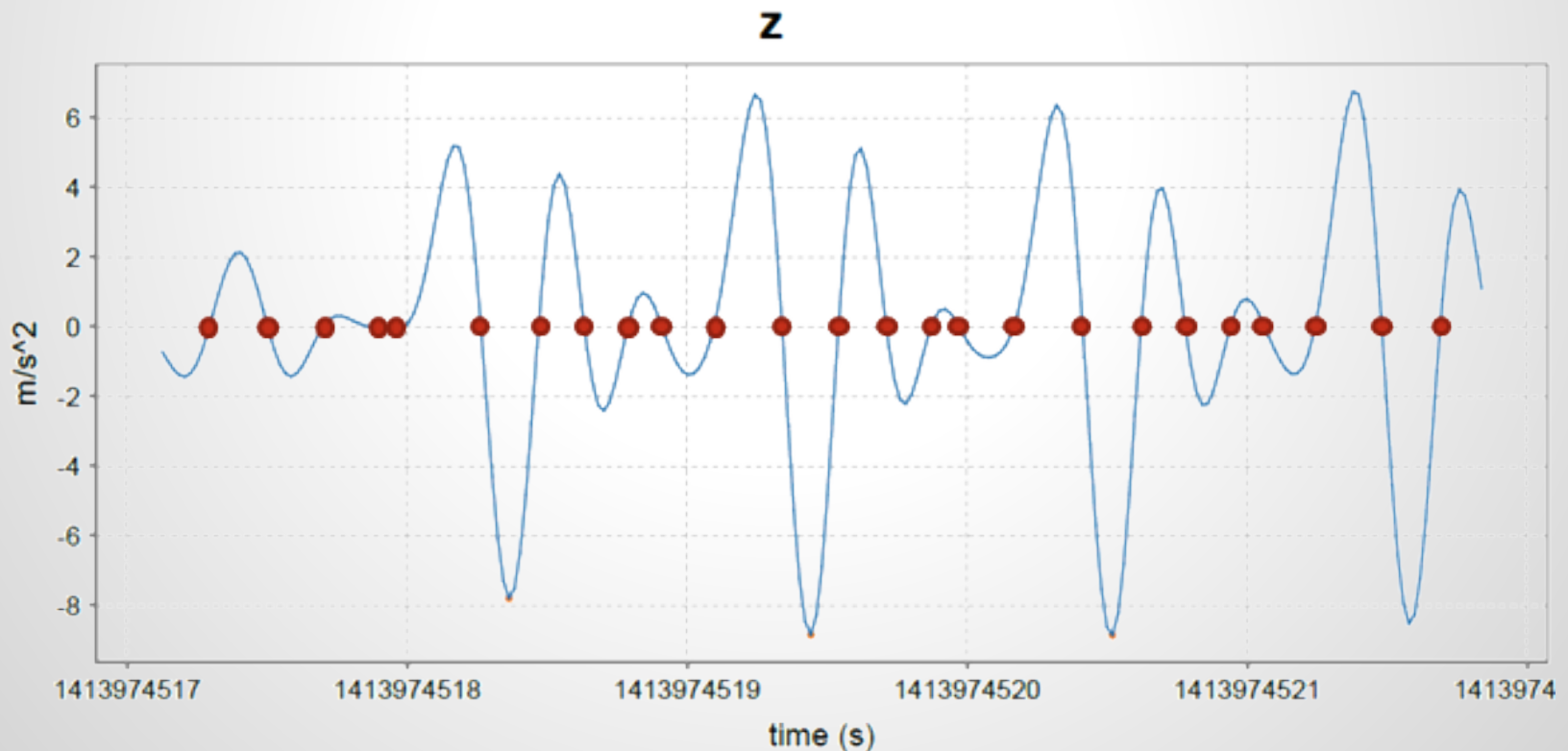
Maat voor “piekvormigheid” van een curve



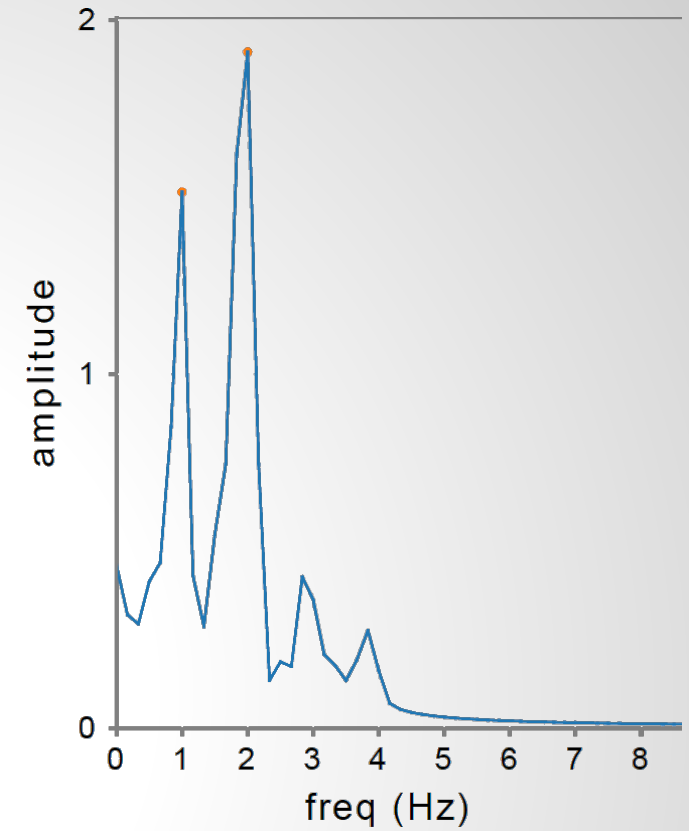
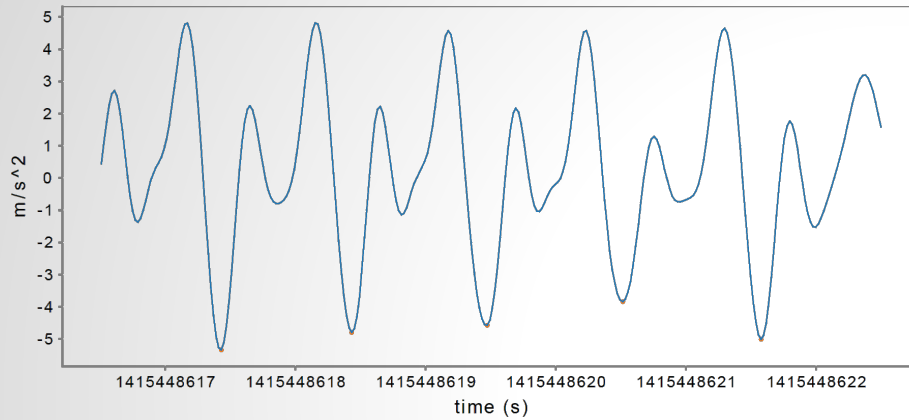
kurtosis = 0,436

Zero crossing rate

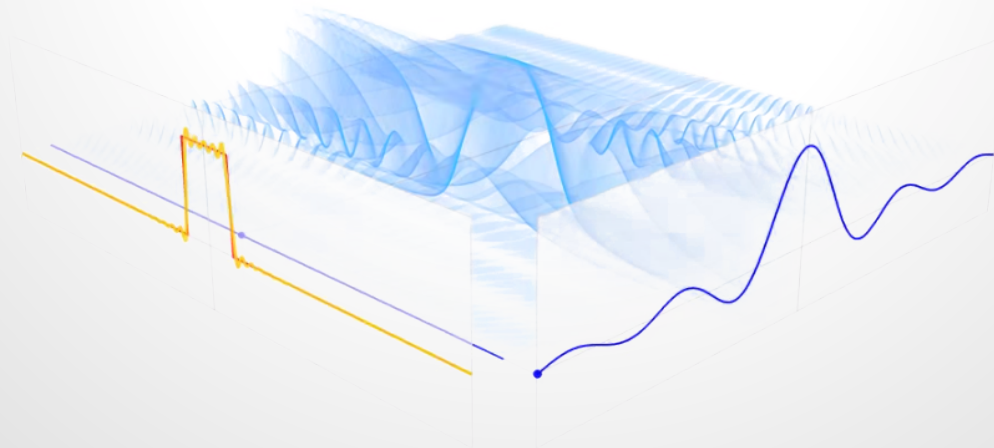
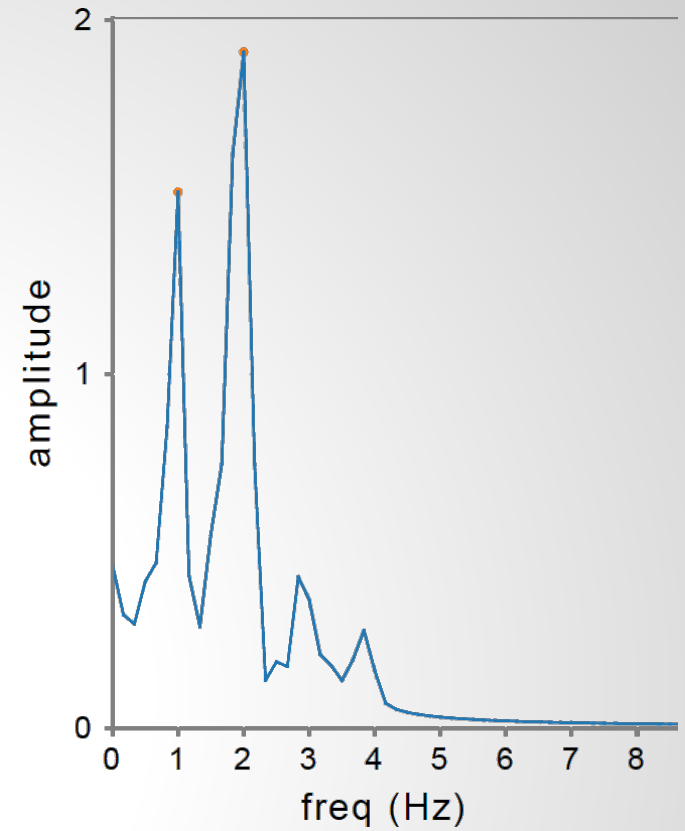
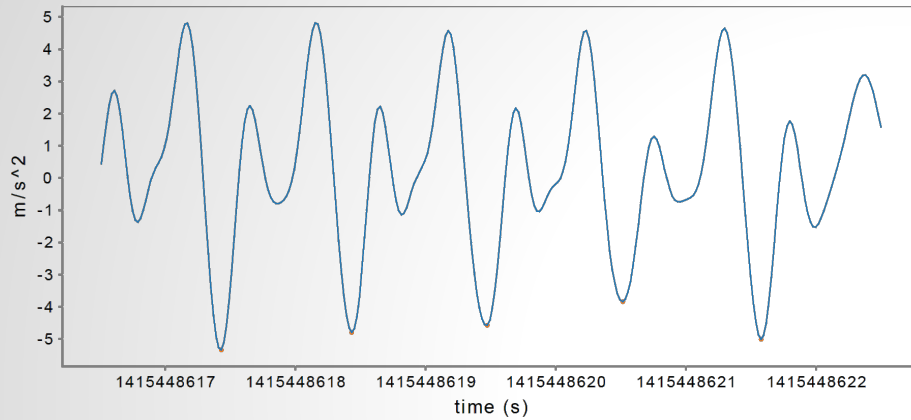
Aantal keer dat een signaal van teken verandert per tijdseenheid



Fourier-transformatie



Fourier-transformatie



Training set

Features

<i>Gemiddelde z-versnelling</i>	<i>Standaardafwijking z-versnelling</i>	...	Label
-0.023672044480712355	2.7266547308816724		wandelen
0.12217549673848728	2.9093965146597998		wandelen
0.15819798154627865	2.5861038302560972		trap op
...



Machine Learning technieken

model berekenen om nieuwe gegevens
te classificeren

(Beschikbaar in Weka) 17

Machine Learning technieken

Base-level classifiers:

- k-Nearest Neighbors (kNN)
- Naive Bayes
- Beslissingsbomen (C4.5)

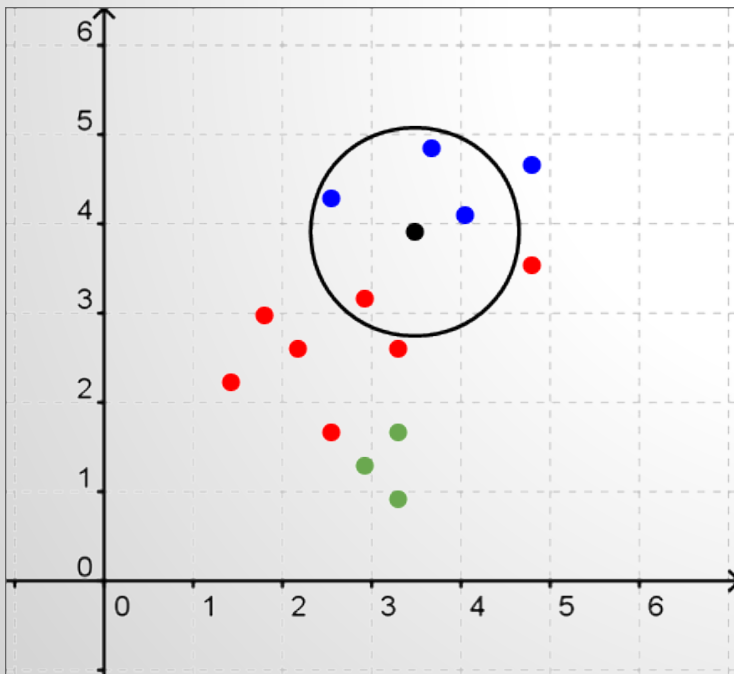
Meta-level classifiers

k-Nearest Neighbors (kNN)

Voorbeeld: $k=4$

- 3 blauw
- 1 rood

⇒ geclassificeerd als blauw

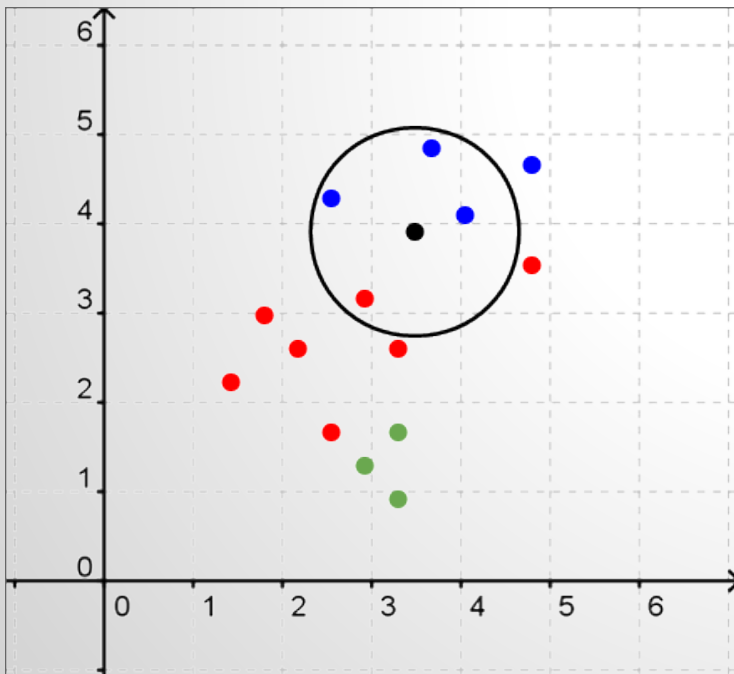


k-Nearest Neighbors (kNN)

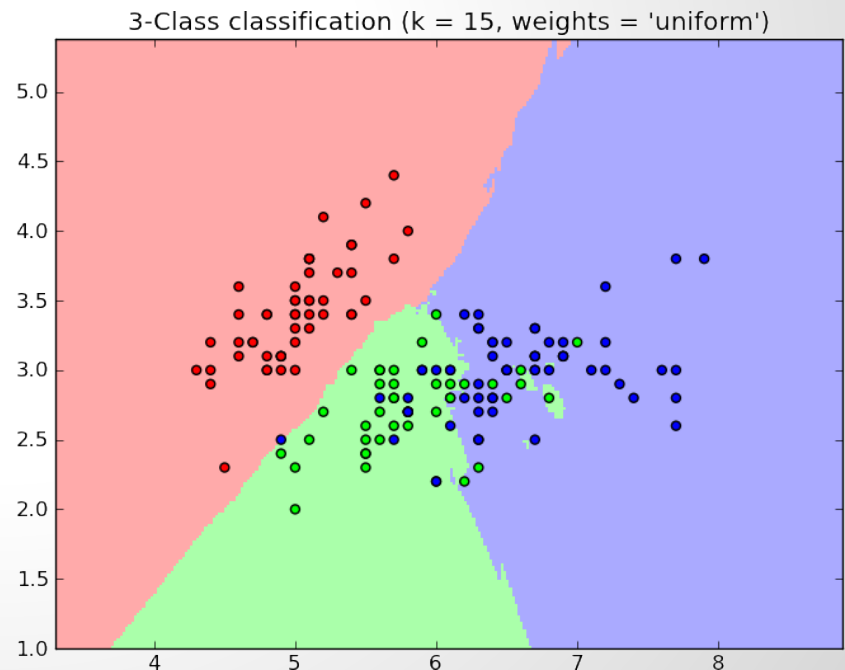
Voorbeeld: $k=4$

- 3 blauw
- 1 rood

⇒ geclassificeerd als blauw



Gebieden bepalen waartoe elk element behoort:
Voronoi-diagram berekenen



Naive Bayes

= Probabilistische classifier

- waarden voor features F_1, \dots, F_n
- label C

$$\begin{aligned} P(C|F_1, \dots, F_n) &= \frac{P(C)P(F_1, \dots, F_n|C)}{P(F_1, \dots, F_n)} \\ &= \frac{P(C)P(F_1|C) \dots P(F_n|C)}{P(F_1, \dots, F_n)} \end{aligned}$$

Naive Bayes

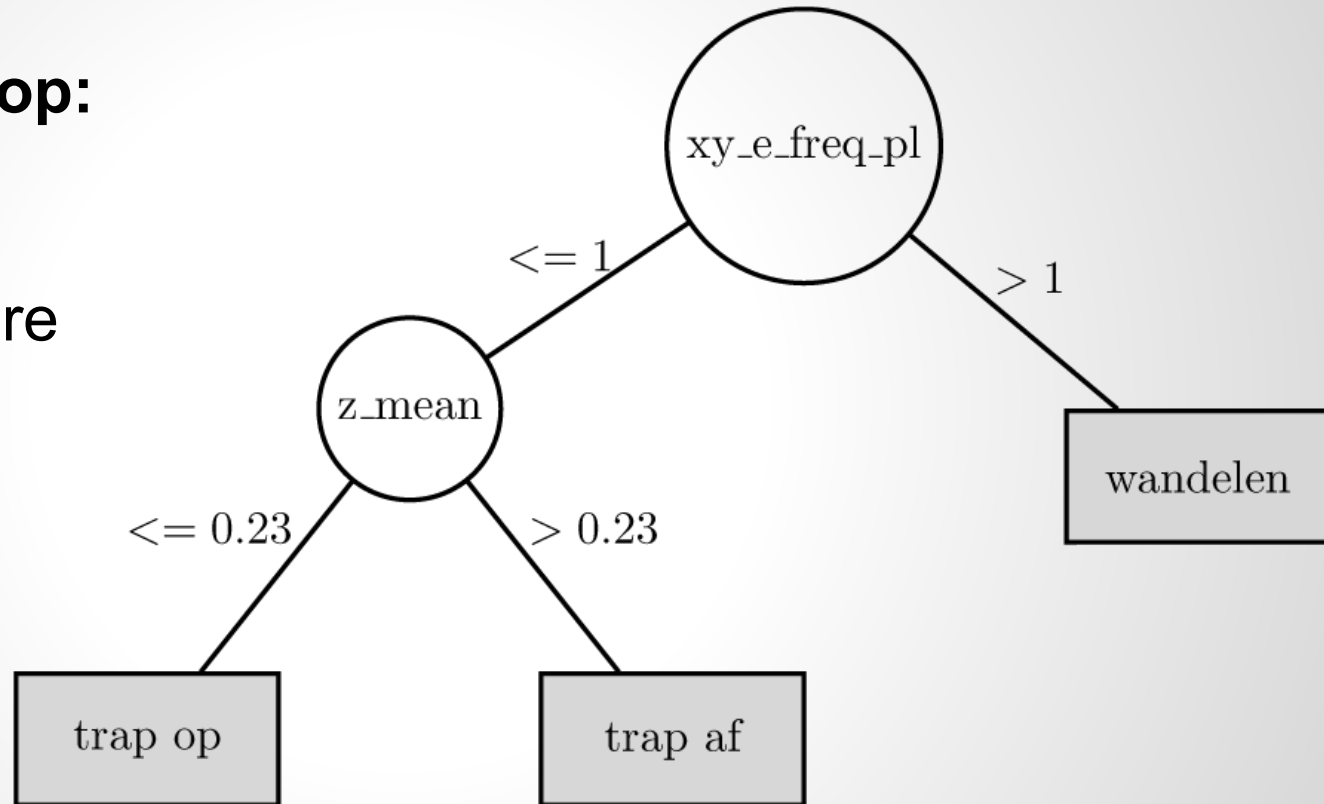
Nieuwe meting met waarden voor features ($\mathbf{f}_1, \dots, \mathbf{f}_n$)
 \Rightarrow geclassificeerd als label \mathbf{c} met de grootste kans:

classificeer(f_1, \dots, f_n)

$$= \max_c P(C = c) P(F_1 = f_1 | C = c) \dots P(F_n = f_n | C = c)$$

Beslissingsbomen

- **Interne knoop:**
feature
- **Tak:**
test op feature
- **Bladknoop:**
label



Hoe opstellen?
 \Rightarrow C4.5 algoritme

C4.5 algoritme: voorbeeld

Training set **T**:

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
2.726	0.051	wandelen
2.909	0.063	wandelen
4.171	0.187	trap af
4.415	0.044	trap af

C4.5 algoritme: voorbeeld

Kies een feature om **T** op te splitsen: 2 mogelijkheden

- *z_stddev*:

≤ 2.909

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
2.726	0.051	wandelen
2.909	0.063	wandelen

> 2.909

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
4.171	0.187	trap af
4.415	0.044	trap af

C4.5 algoritme: voorbeeld

Kies een feature om **T** op te splitsen: 2 mogelijkheden

- *corr_z_xy_e*:

≤ 0.051

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
4.415	0.044	trap af
2.726	0.051	wandelen

> 0.051

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
2.909	0.063	wandelen
4.171	0.187	trap af

C4.5 algoritme: voorbeeld

*Welke feature **F** kiezen voor opsplitsing?*

- Bereken de **entropy** van elke deelverzameling T_i

$$entropy(T_i) = - \sum_{j=1}^k \frac{freq(C_j, T_i)}{|T_i|} \times \log_2 \frac{freq(C_j, T_i)}{|T_i|}$$

- alle elementen van T_i zelfde label $\Rightarrow entropy(T_i) = 0$
- alle elementen van T_i ander label $\Rightarrow entropy(T_i) = 1$

C4.5 algoritme: voorbeeld

*Welke feature **F** kiezen voor opsplitsing?*

- Bereken **entropy_F** van opsplitsing van **T** volgens **F**

$$entropy_F(T) = \sum_{i=1}^n \frac{|T_i|}{|T|} \times entropy(T_i)$$

hierin is

$$entropy(T_i) = - \sum_{j=1}^k \frac{freq(C_j, T_i)}{|T_i|} \times \log_2 \frac{freq(C_j, T_i)}{|T_i|}$$

C4.5 algoritme: voorbeeld

Welke feature F kiezen voor opsplitsing?

- Kies de feature met hoogste **gain**

$$gain(F) = entropy(T) - entropy_F(T)$$

C4.5 algoritme: voorbeeld

Kies een feature om **T** op te splitsen: 2 mogelijkheden

- *z_stddev*: **gain(z_stddev) = 1**

≤ 2.909

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
2.726	0.051	wandelen
2.909	0.063	wandelen

> 2.909

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
4.171	0.187	trap af
4.415	0.044	trap af

C4.5 algoritme: voorbeeld

Kies een feature om **T** op te splitsen: 2 mogelijkheden

- *corr_z_xy_e*: **gain(corr_z_xy_e) = 0**

≤ 0.051

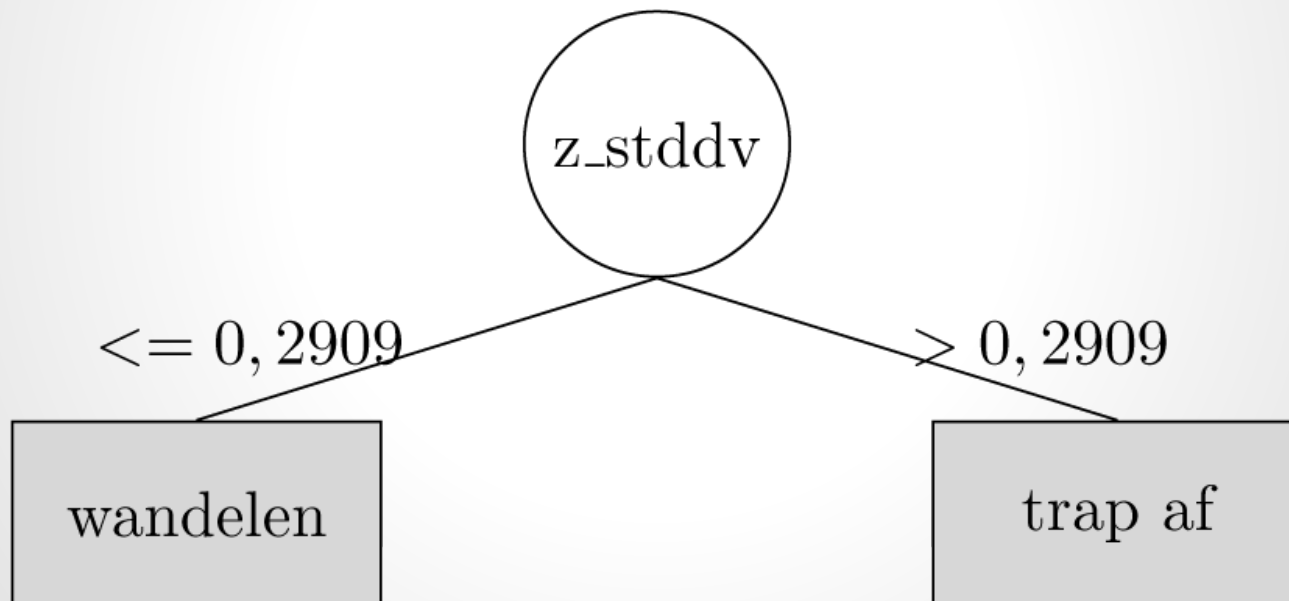
<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
4.415	0.044	trap af
2.726	0.051	wandelen

> 0.051

<i>z_stddev</i>	<i>corr_z_xy_e</i>	Label
2.909	0.063	wandelen
4.171	0.187	trap af

C4.5 algoritme: voorbeeld

De resulterende beslissingsboom:



Besluit

1. Gegevens opmeten:
accelerometer + gyroscoop
2. Features berekenen
3. Machine Learning
training set \Rightarrow model \Rightarrow nieuwe gegevens classificeren

Vragen?