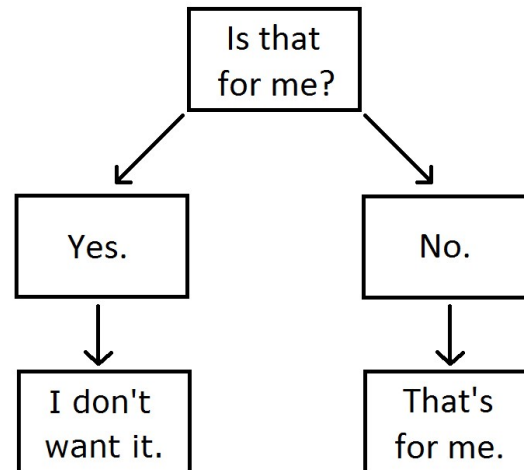


Data-Science 1

beslissingsbomen

My Cat's Decision-Making Tree.



Inhoud

- voorbeeld: ad eater
- voorbeeld: Simpsons
- het ID3 algoritme
- praktisch: orange
- bossen

Voorbeeld:
ad eater

Ad eater

- webpagina's bevatten images
- sommige images zijn reclame, andere niet
- kan ik automatisch detecteren wat reclame is?



Ad eater

- probleem: aan de hand van welke parameters kan je bepalen of een beeld reclame is?
 - afmetingen, positie, kleur
 - html attributen
 - alt tag
 - url van het beeld
 - url van de link (als clickable)
 - ...

Ad eater

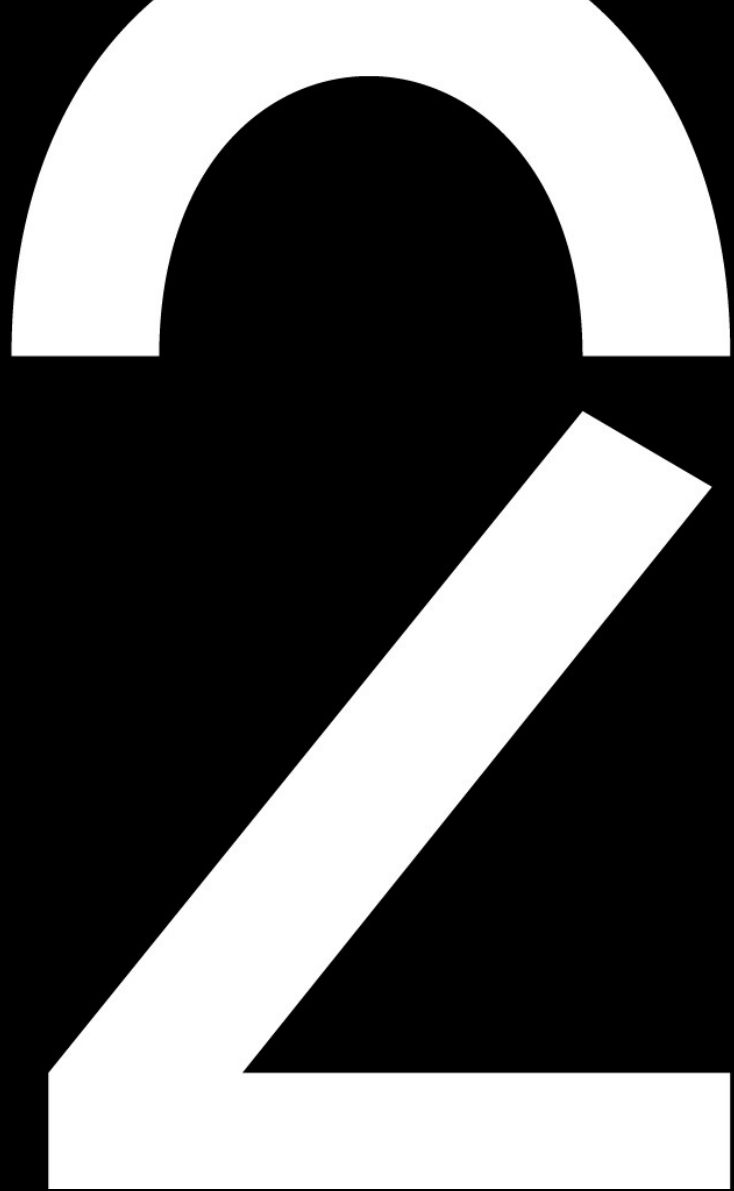
- oplossing
 - zoek een aantal voorbeelden (3279)
 - som alle eigenschappen op (1558)
 - zet alles in een tabel met 3279 rijen en 1558 kolommen
 - bepaal handmatig of deze voorbeelden reclame zijn of niet
 - we kunnen niet alle gevallen opsommen, dus laat computer hieruit "leren"










Ad eater

- resultaat: regels
 - als
 - aspect ratio > 4.5833
 - alt doesn't contain "to"
 - alt contains "click+here"
 - url doesn't contain "http+www"
 - dan: reclame!

voordeel: de computer zoekt zelf van welke variabelen de target afhankelijk is!

Voorbeeld:
Simpsons



	Naam	Haarlengte (inch)	Gewicht (lbs)	Leeftijd (jaren)	Geslacht
	Homer	0	250	36	M
	Marge	10	150	34	V
	Bart	2	90	10	M
	Lisa	6	78	8	V
	Maggie	4	20	1	V
	Abe	1	170	70	M
	Selma	8	160	41	V
	Otto	10	180	38	M
	Krusty	6	200	45	M



Comic

8











290

38

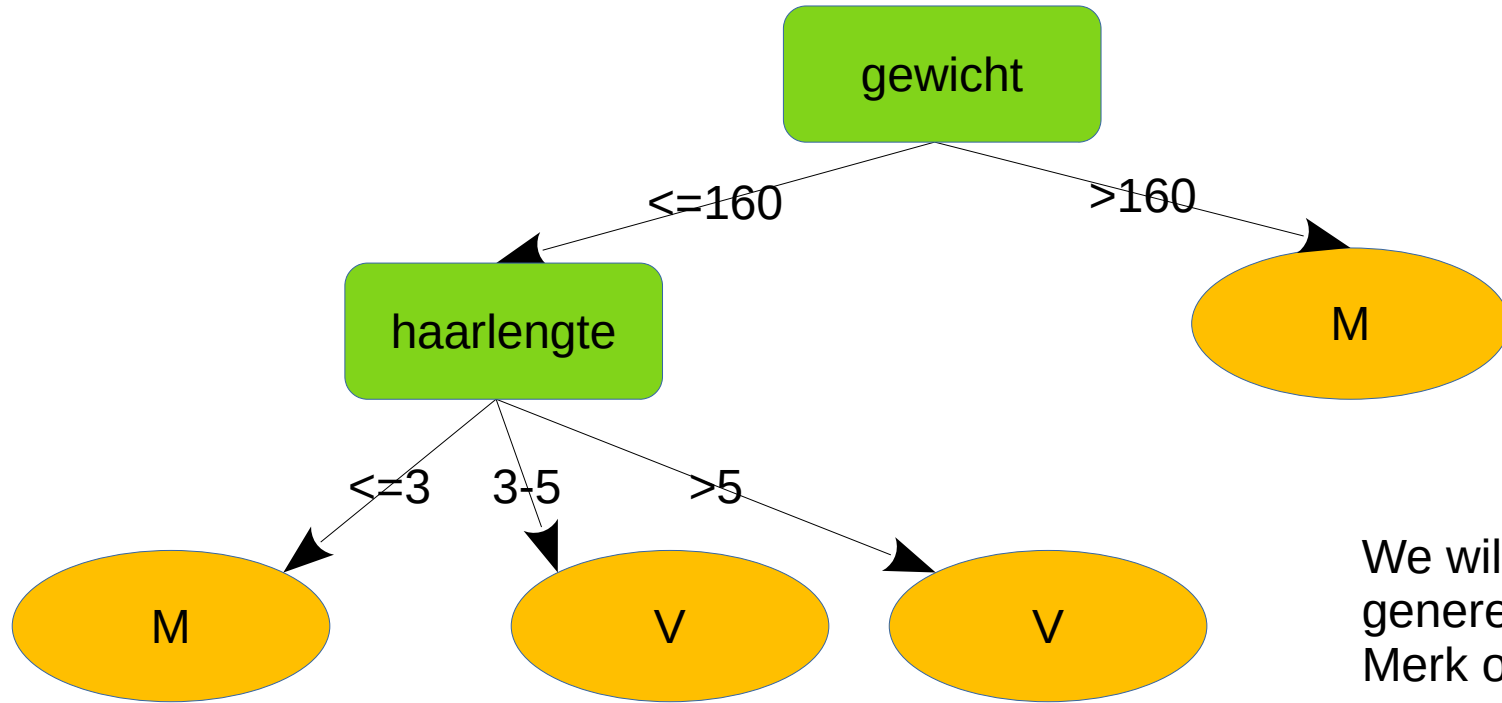
??

Stap 1: discretiseren

- variabelen moeten discreet zijn (continue variabelen zien we later)
- niet te veel verschillende waarden
- we zetten alles om naar nominaal en ordinaal meetniveau
 - hoe kan je een continue variabele omzetten naar een discrete met ordinaal meetniveau? (hint: zie frequenties)

	Naam	Haarlengte (inch)	Gewicht (lbs)	Leeftijd (jaren)	Geslacht
	Homer	≤ 3	> 160	30-40	M
	Marge	> 5	≤ 160	30-40	V
	Bart	≤ 3	≤ 160	≤ 30	M
	Lisa	> 5	≤ 160	≤ 30	V
	Maggie	3-5	≤ 160	≤ 30	V
	Abe	≤ 3	> 160	> 40	M
	Selma	> 5	≤ 160	> 40	V
	Otto	> 5	> 160	30-40	M
	Krusty	> 5	> 160	> 40	M
	Comic	> 5	> 160	30-40	??

Dit zoeken we:



We willen dit laten genereren.
Merk op: leeftijd speelt geen rol!

Algemeen

- beslissingsbomen zijn een voorbeeld van supervised learning
- het is een classificatietechniek
- (ze kunnen ook voor regressie gebruikt worden, maar dat is buiten de scope van deze cursus)

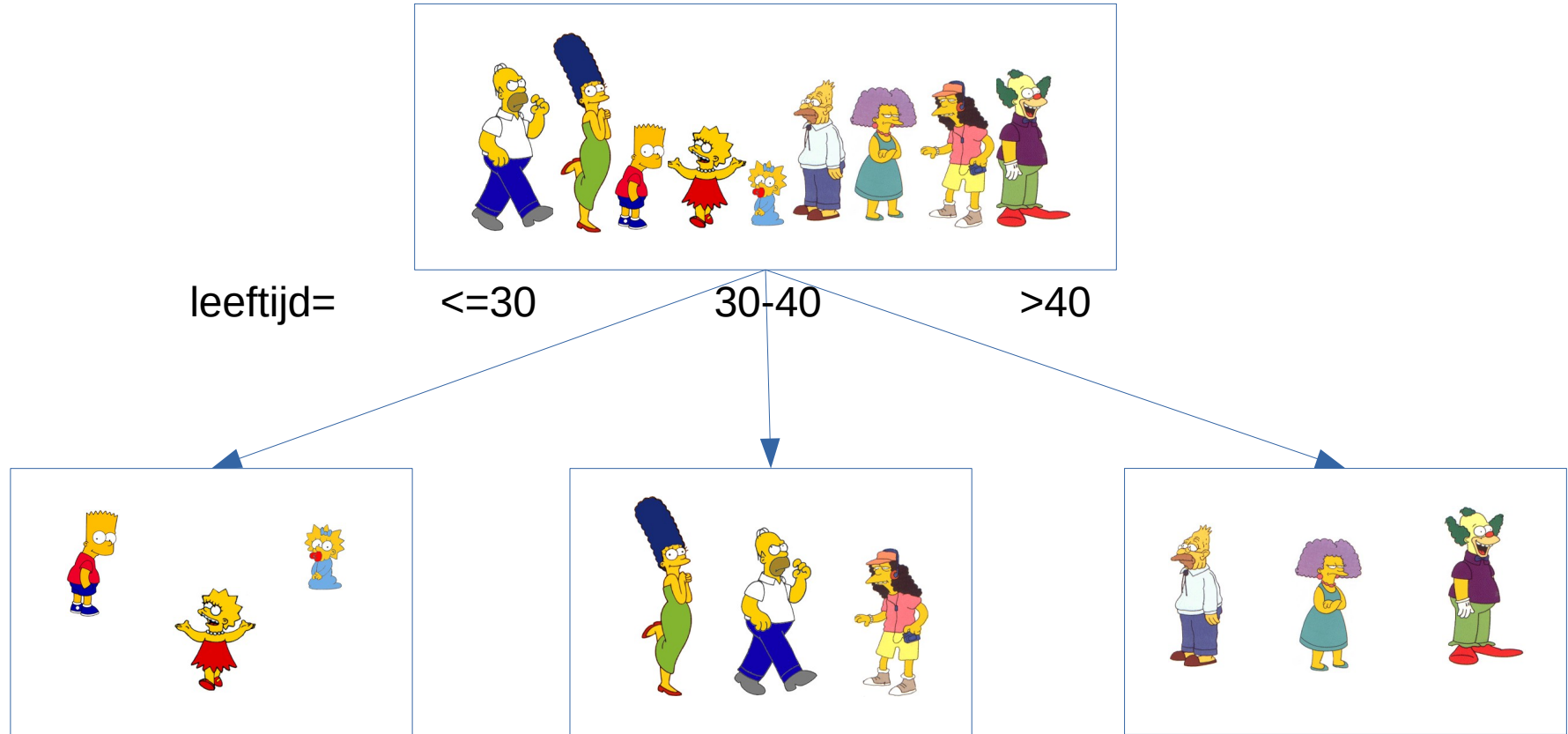
ID3

Hoe een boomstructuur vinden?

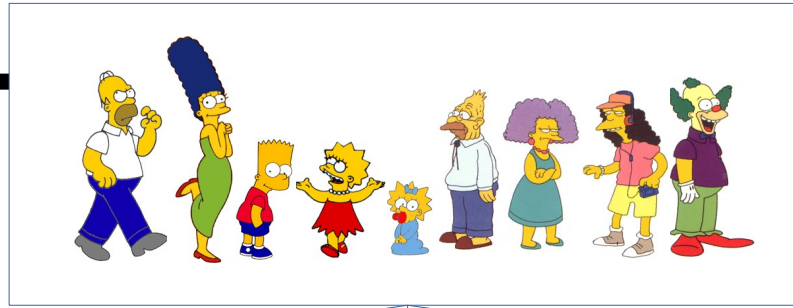
- kies een kolom
- maak een node voor deze kolom
- voor iedere mogelijke waarde in de gekozen kolom:
 - maak een “kind-node” met de naam van de waarde
 - selecteer enkel de lijnen waar de waarde voorkomt
 - zet deze lijnen in een subtabel
 - als alle te voorspellen waarden in de subtabel gelijk zijn, dan stop je
 - voer het algoritme recursief uit voor deze subtabel

Voorbeeld Simpsons

- kies kolom "leeftijd"
- leeftijd heeft 3 mogelijkheden: ≤ 30 , 30-40 of > 40
- maak dus een node "leeftijd" met 3 kinderen
- bereken voor het eerste kind een tabel met alle rijen waarbij leeftijd ≤ 30
- bereken voor het tweede kind een tabel met alle rijen waarbij leeftijd 30-40
- bereken voor het derde kind een tabel met alle rijen waarbij leeftijd > 40



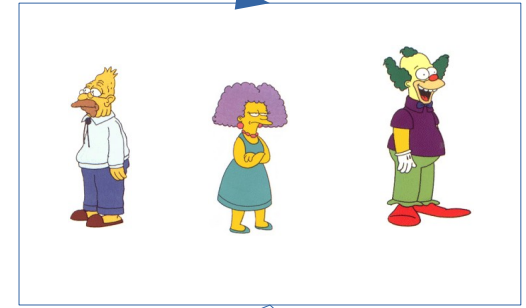
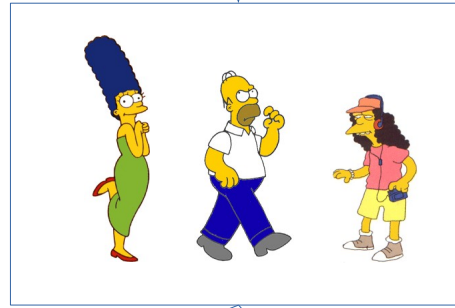
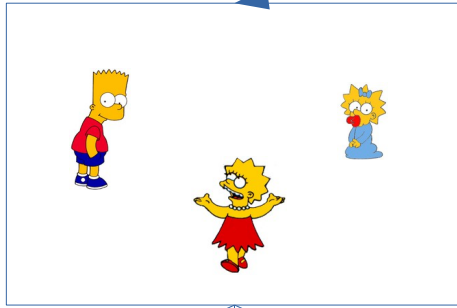
overal zitten nog mannen en vrouwen



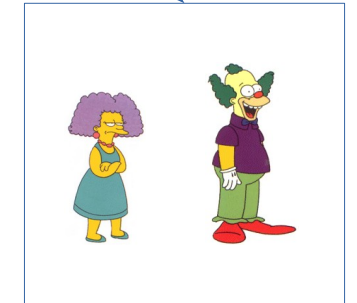
leeftijd=

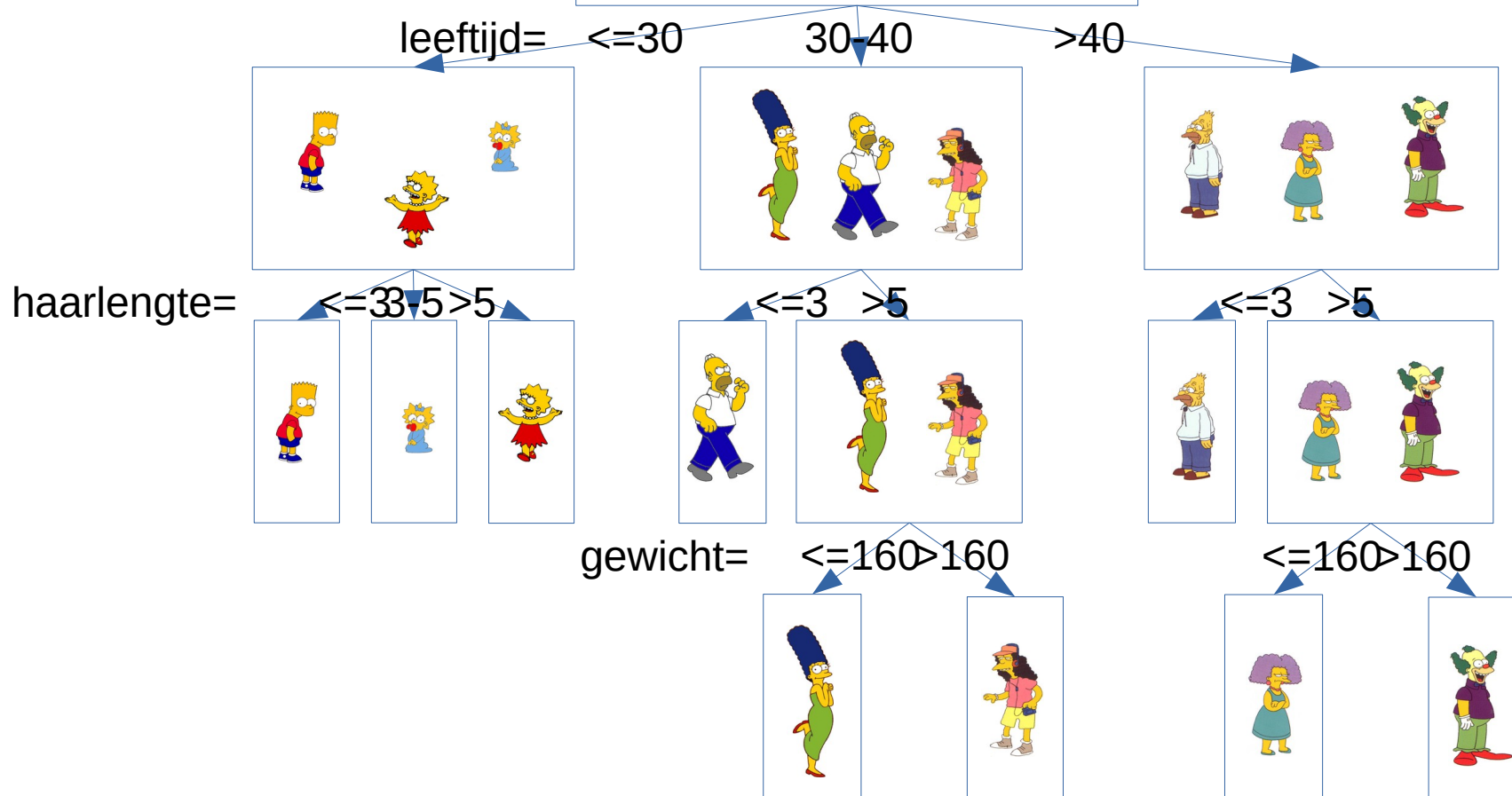
 ≤ 30

30-40

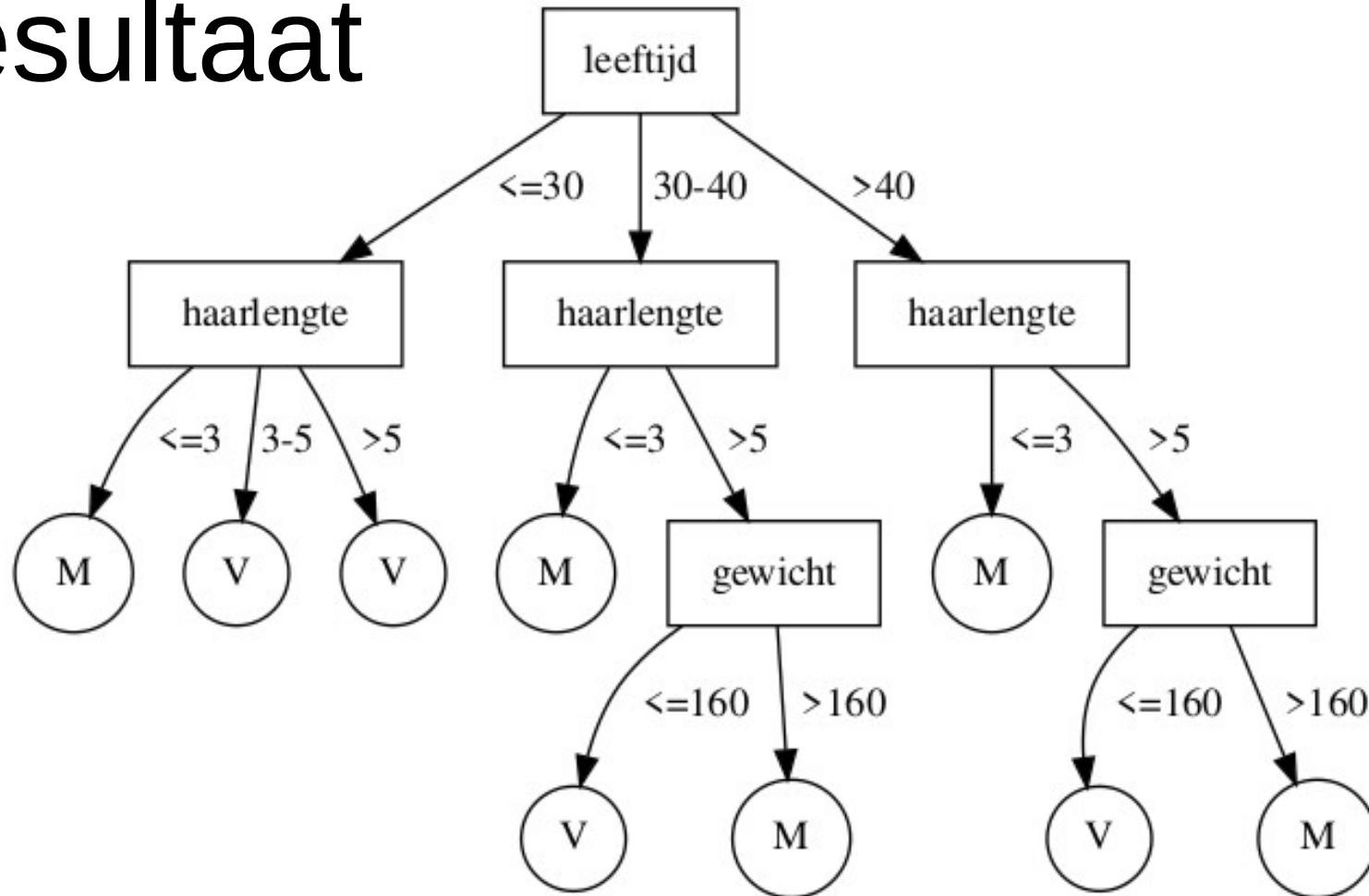
 > 40 

haarlengte=

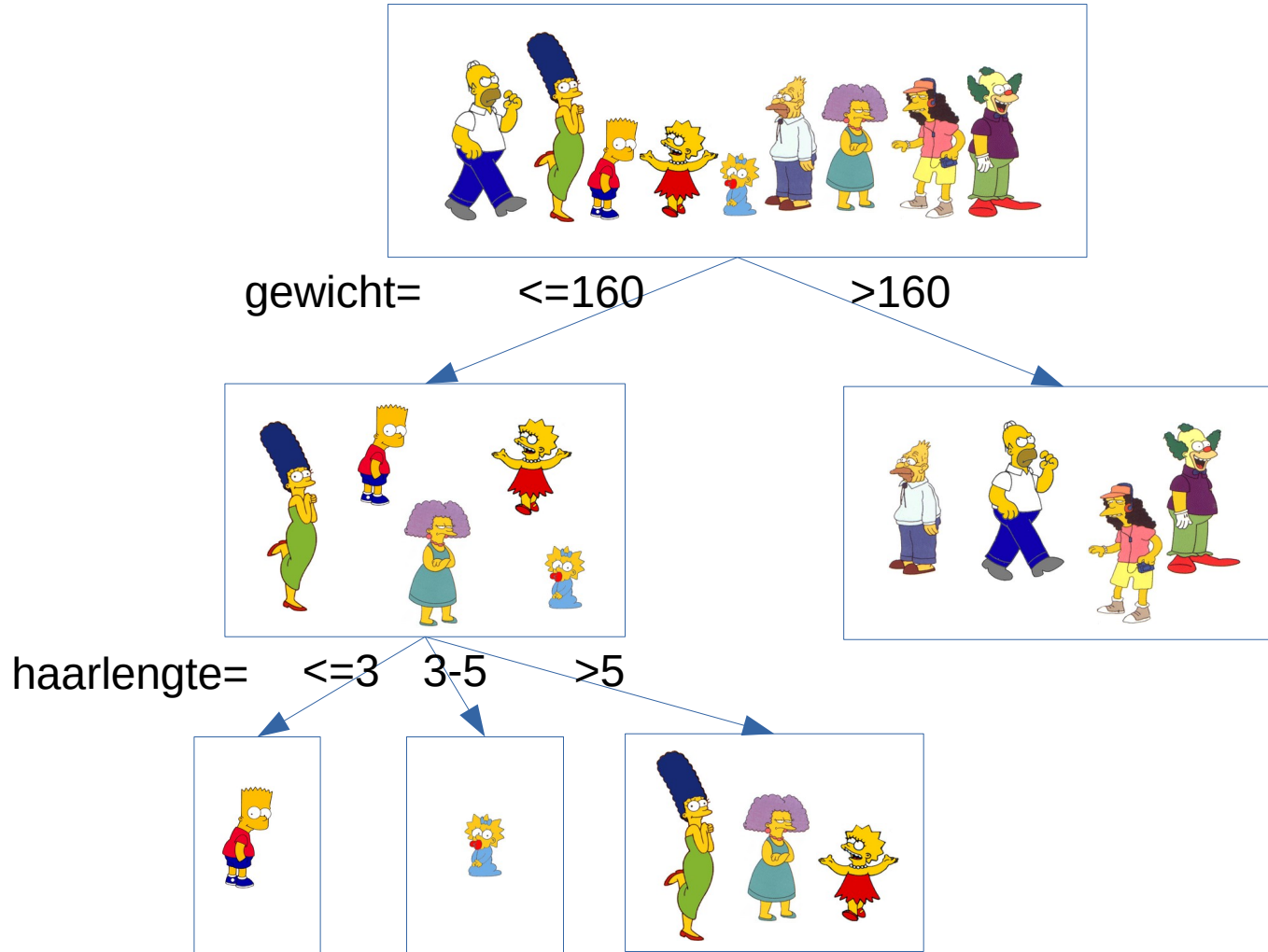
 ≤ 3 3-5 > 5 ≤ 3 > 5 ≤ 3 > 5 



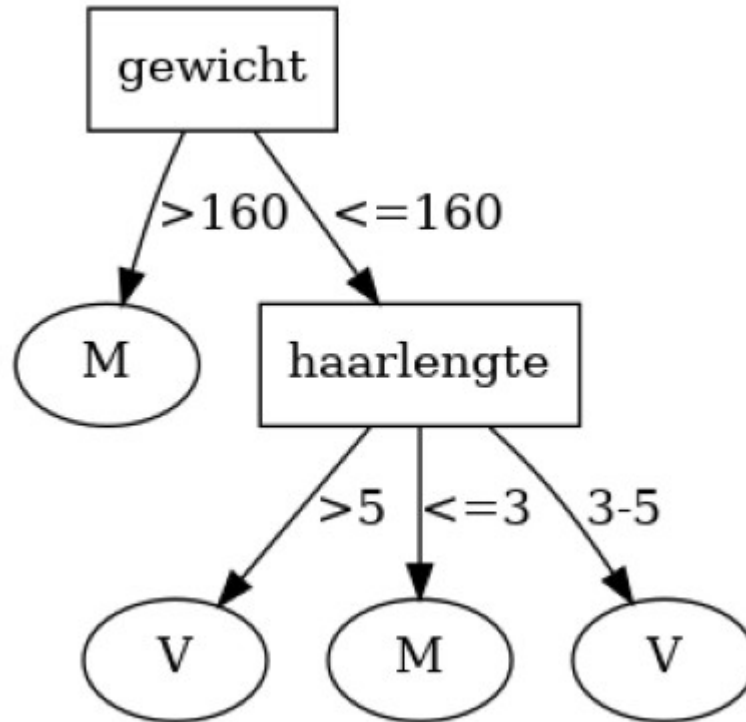
Resultaat



Als we begonnen met gewicht^{22/50}



Resultaat



Keuze van de kolom

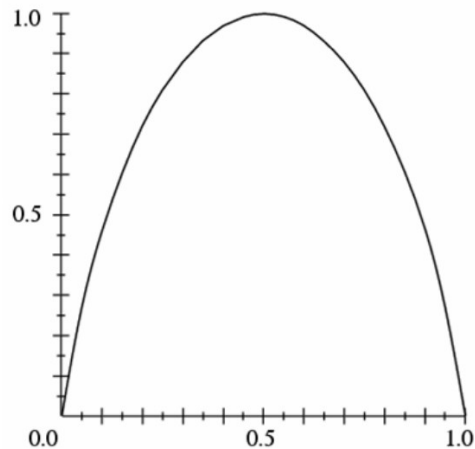
- waarom begonnen we met "leeftijd"?
 - “gewicht” zou beter geweest zijn
- hoe kunnen we dit weten?
- we zoeken een zo klein mogelijke boomstructuur
- we doen dit door de “Entropie” en de “Information gain” te berekenen

Entropie

- maat voor "chaos" in de kolom met resultaten (de "afhankelijke variabele" of "target")
- allemaal dezelfde waarde: $E(\text{target})=0$
- formule:

$$E(\text{target}) = \sum_{\text{target waarden}} -(p/n) \cdot \log_2(p/n)$$

- p = aantal rijen met de gegeven waarde in de target kolom
 - n = aantal rijen in de tabel
 - p/n is dus de relatieve frequentie van de waarde in de target kolom
- voorbeeld (simpsons heeft 4 vrouwen en 5 mannen):
 $E(\text{simpsons}) = -(4/9) \cdot \log_2(4/9) - (5/9) \cdot \log_2(5/9)$



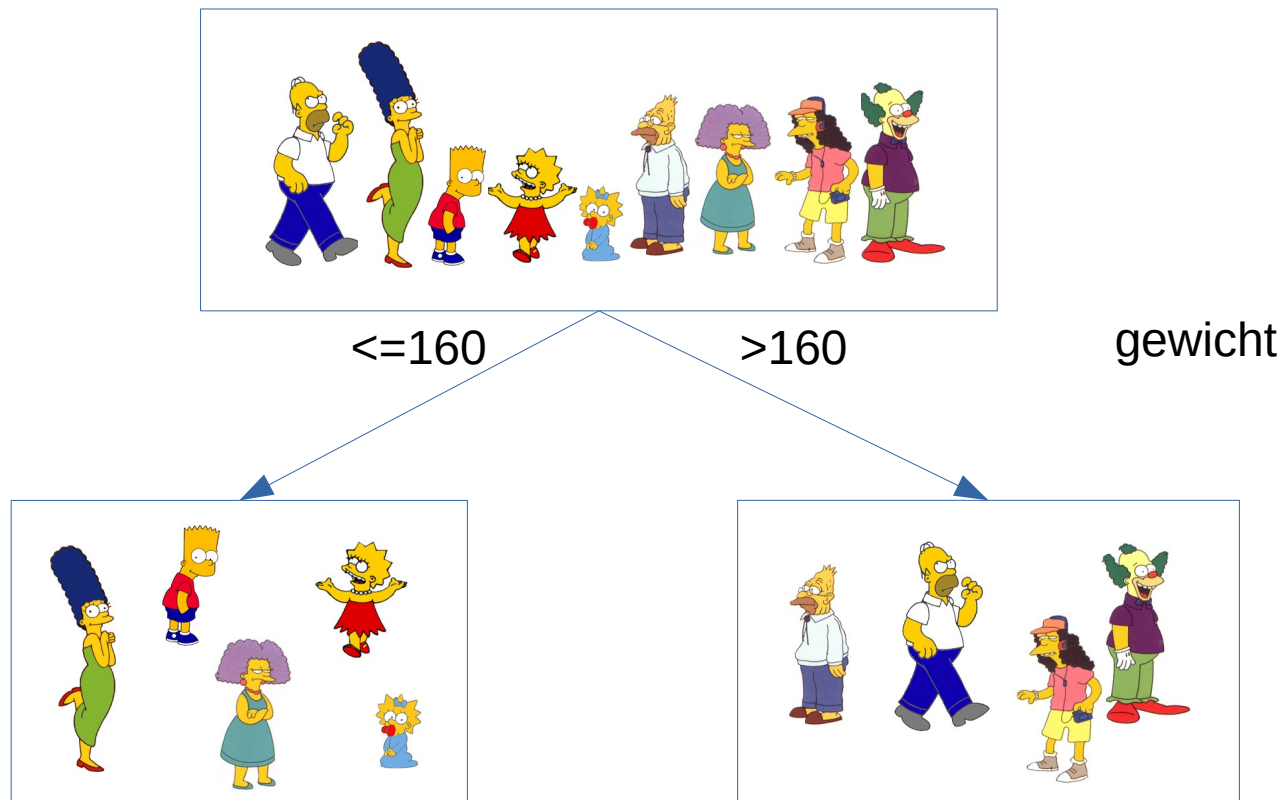
Keuze van de kolom

- zoek de kolom met het grootste “onderscheidend vermogen”
- men noemt dit "**information gain**"
- vergelijk met correlatie maar dan voor nominale variabelen
- $Gain(kolom) = E(target) - \sum_{\text{waarden van kolom}} (p/n) \cdot E(target_{\text{subtabel}})$
 - “subtabel” bevat enkel een bepaalde waarde voor de gegeven kolom
 - p is het aantal rijen in de subtabel
 - n is het totaal aantal rijen in de tabel
 - p/n is dus de relatieve frequentie van de waarde in de kolom
 - E(tabel) is de entropie van die tabel

$$\text{Gain}(\text{gewicht}) = 0,991 - (5/9) \cdot 0,722 - (4/9) \cdot 0 = 0,59$$

27/50

$$E(\text{simpsons}) = -(4/9) \cdot \log_2(4/9) - (5/9) \cdot \log_2(5/9) = 0,991$$

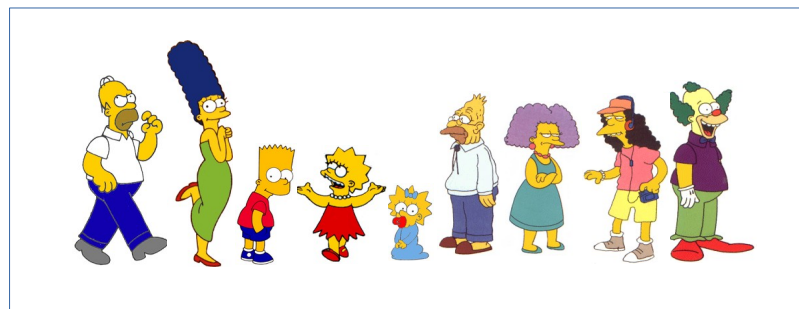


$$E(\text{gewicht} \leq 160) = -(4/5) \cdot \log_2(4/5) - (1/5) \cdot \log_2(1/5) = 0,722$$

$$E(\text{gewicht} > 160) = -(0/4) \cdot \log_2(0/4) - (4/4) \cdot \log_2(4/4) = 0$$

$$\text{Gain}(\text{haarlengte}) = 0,991 - (3/9) \cdot 0 - (1/9) \cdot 0 - (5/9) \cdot 0,971 = 0,452_{28/50}$$

$$E(\text{simpsons}) = -(4/9) \cdot \log_2(4/9) - (5/9) \cdot \log_2(5/9) = 0,991$$

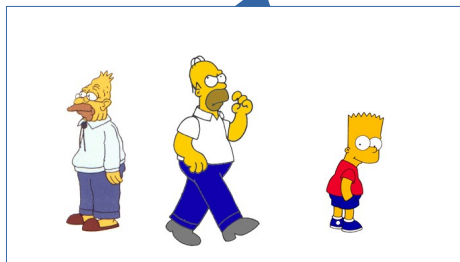


≤ 3

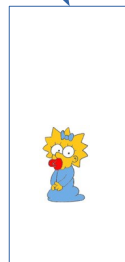
$> 3 \text{ en } \leq 5$

> 5

haarlengte



$$E(\text{haarlengte} \leq 3) = 0$$



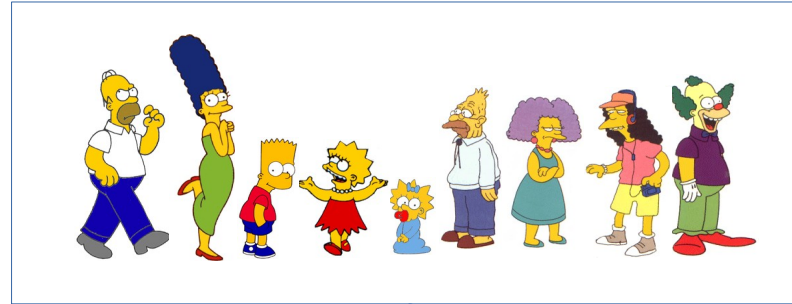
$$E(3 < \text{haarlengte} \leq 5) = 0$$



$$E(\text{haarlengte} > 5) = -(2/5) \cdot \log_2(2/5) - (3/5) \cdot \log_2(3/5) = 0,971$$

$$Gain(leeftijd) = \dots - (\dots/9) \cdot \dots - (\dots/9) \cdot \dots - (\dots/9) \cdot \dots = \dots \quad 29/50$$

$$E(\text{simpsons}) = -(4/9) \cdot \log_2(4/9) - (5/9) \cdot \log_2(5/9) = 0,991$$



≤ 30

> 30 en ≤ 40

> 40

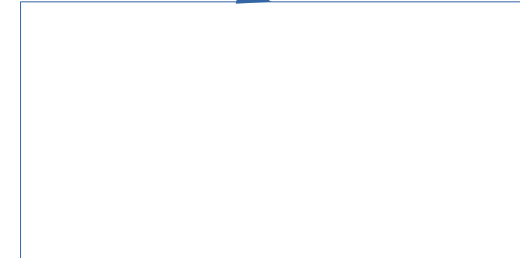
leeftijd



$$E(\text{leeftijd} \leq 30) = \dots$$



$$E(30 < \text{leeftijd} \leq 40) = \dots$$

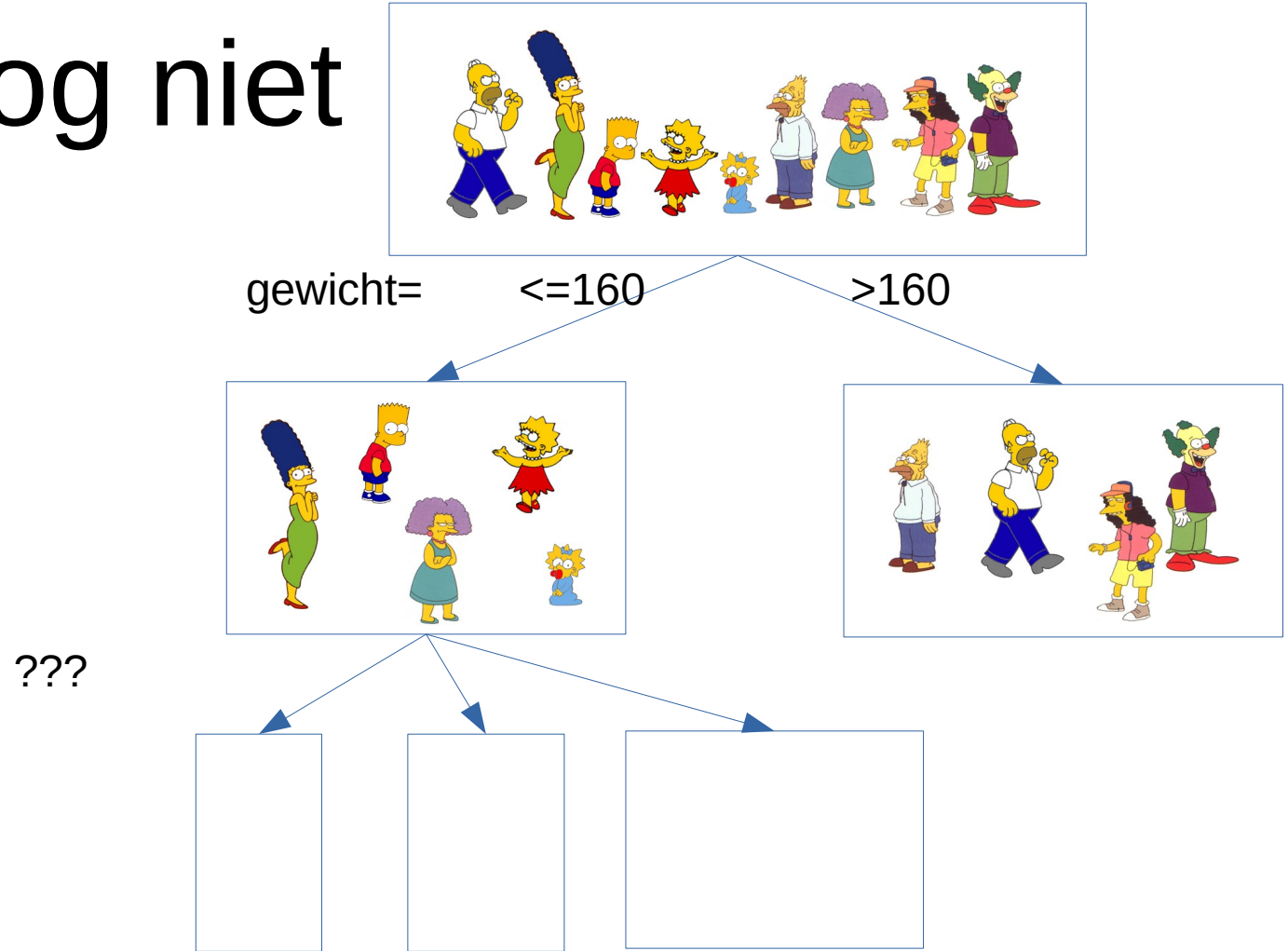


$$E(\text{leeftijd} > 40) = \dots$$

Samenvatting Gains

- we bekomen dus volgende gains:
 - $\text{Gain}(\text{haarlengte}) = 0,452$
 - $\text{Gain}(\text{gewicht}) = 0,590$
 - $\text{Gain}(\text{leeftijd}) = 0,073$
- gewicht heeft hoogste gain => kies dit als eerste kolom
- opmerkingen
 - $\text{Gain}(\text{geslacht}) = 0,991$
 - hoogst mogelijke gain = entropie
 - $\text{Gain}(\text{naam}) = ?$

We zijn nog niet klaar...



Andere algoritmes

- andere algoritmes kunnen:
 - continue variabelen automatisch opsplitsen
 - ze zoeken een “split point” voor iedere kolom zodat de gain het hoogst is
 - omgaan met ontbrekende waarden
 - de boom afkappen indien te complex (“pruning”)
 - iedere uitkomst wordt nu weergegeven met een “waarschijnlijkheid”
- voorbeelden: C4.5, J48, CART, ...

Orange

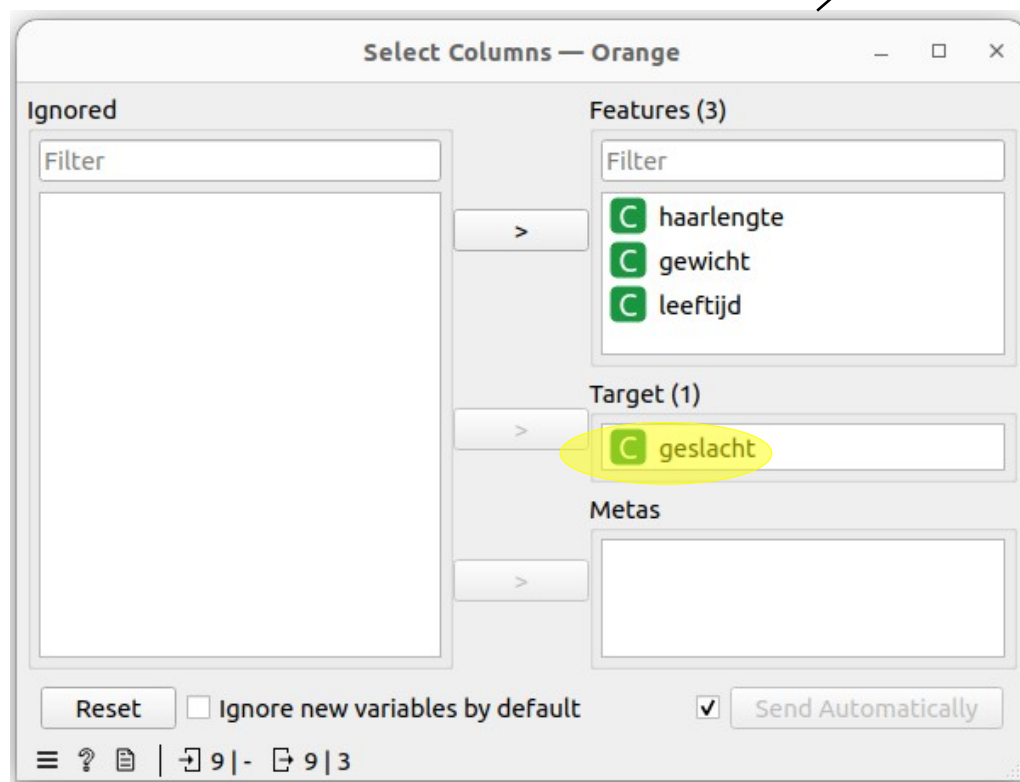


Orange



	1	2	3	4
1	haarlengte	gewicht	leeftijd	geslacht
2	<3	>160	30-40	M
3	>5	<=160	30-40	V
4	<3	<=160	<30	M
5	>5	<=160	<30	V
6	3-5	<=160	<30	V
7	<3	>160	>40	M
8	>5	<=160	>40	V
9	>5	>160	30-40	M
10	>5	>160	>40	M

Orange



Orange



Tree — Orange [X]

Name

Tree

Parameters

- ☐ Induce binary tree
- ☐ Min. number of instances in leaves: 2
- ☐ Do not split subsets smaller than: 5
- ☐ Limit the maximal tree depth to: 100

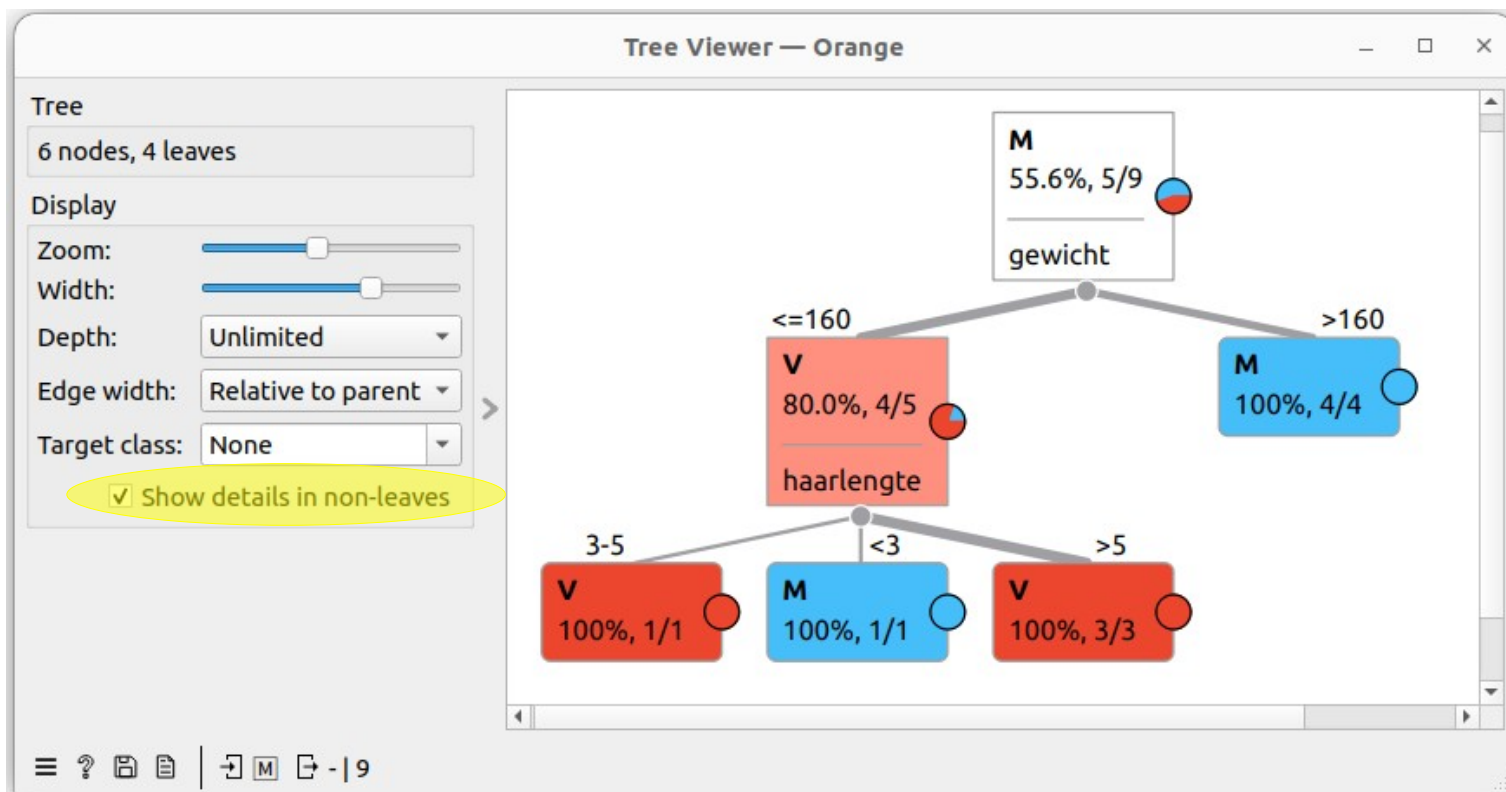
Classification

- ☐ Stop when majority reaches [%]: 95

☒ Apply Automatically

≡ ? [Icon] 9 | - [Icon] [L] [M]

Orange

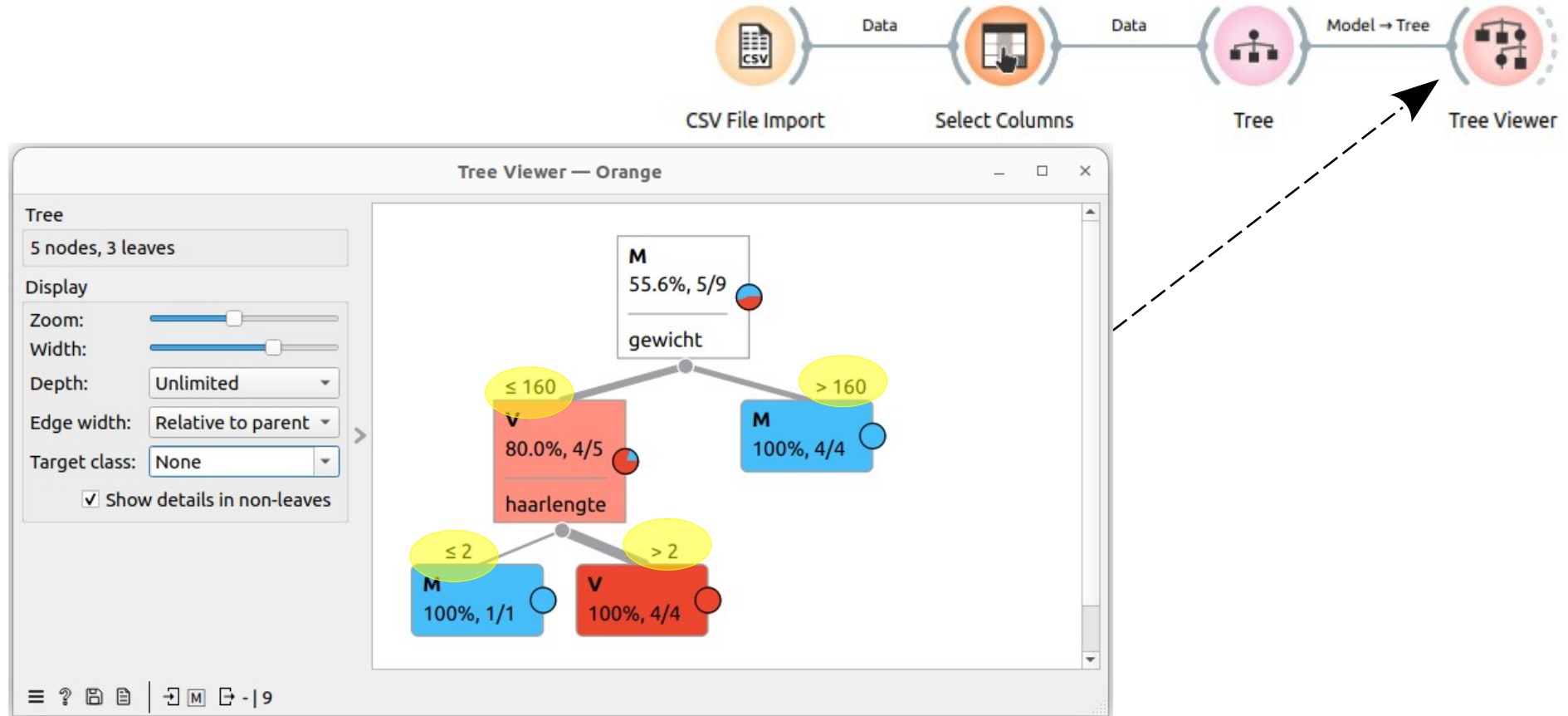


Maar Orange kan meer...



1		N 2	N 3	N 4	C 5
1	naam	haarlengte	gewicht	leeftijd	geslacht
2	Homer	0	250	36	M
3	Marge	10	150	34	V
4	Bart	2	90	10	M
5	Lisa	6	78	8	V
6	Maggie	4	20	1	V
7	Abe	1	170	70	M
8	Selma	8	160	41	V
9	Otto	10	180	38	M
10	Krusty	6	200	45	M

Maar Orange kan meer...



Entropie en information gain



	C 1	C 2	C 3	C 4
1	haarlengte	gewicht	leeftijd	geslacht
2	<3	>160	30-40	M
3	>5	<=160	30-40	V
4	<3	<=160	<30	M
5	>5	<=160	<30	V
6	3-5	<=160	<30	V
7	<3	>160	>40	M
8	>5	<=160	>40	V
9	>5	>160	30-40	M
10	>5	>160	>40	M

dit werkt enkel met
categorische variabelen

Entropie en information gain



Formula — Orange

Variable Definitions

New **D1** geslacht

Remove ☐ Meta attribute Select Feature Select Function

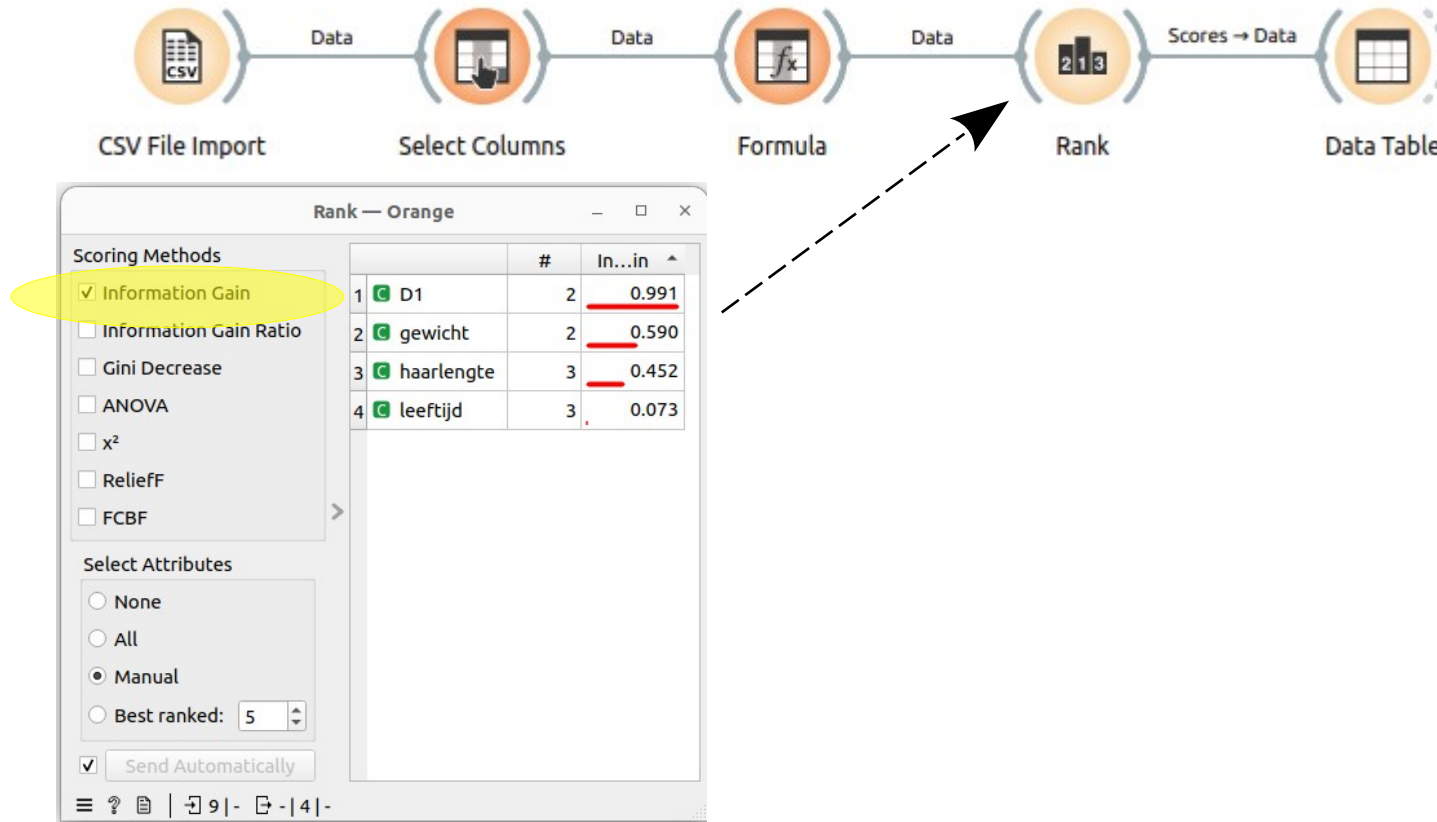
Values (optional) A, B ...

C D1 := geslacht

Send

≡ ? | 9 9

Entropie en information gain

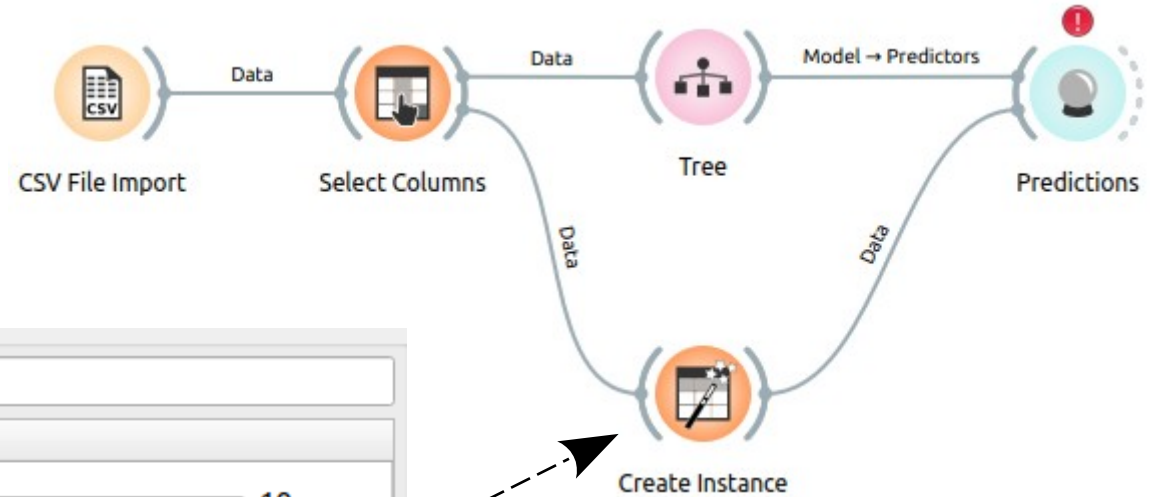


Entropie en information gain



	Feature	Info. gain
1	haarlengte	0.451659
2	gewicht	0.590005
3	leeftijd	0.0727802
4	D1	0.991076

Voorspellen



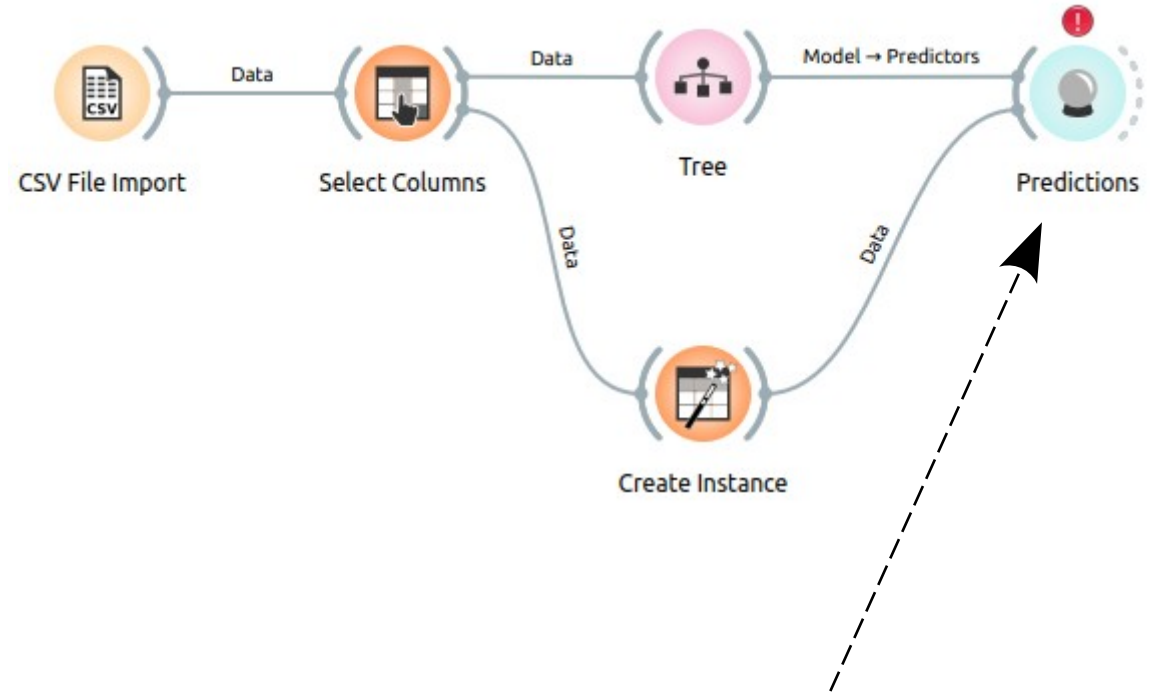
Filter...

Variable	Value
N haarlengte	1,000 <input type="text"/> 0 <input type="range"/> 10
N gewicht	180,000 <input type="text"/> 20 <input type="range"/> 250
N leeftijd	36,000 <input type="text"/> 1 <input type="range"/> 70
C geslacht	? <input type="text"/>

Median Mean Random Input

☐ Append this instance to input data ☒ Apply Automatically

Voorspellen



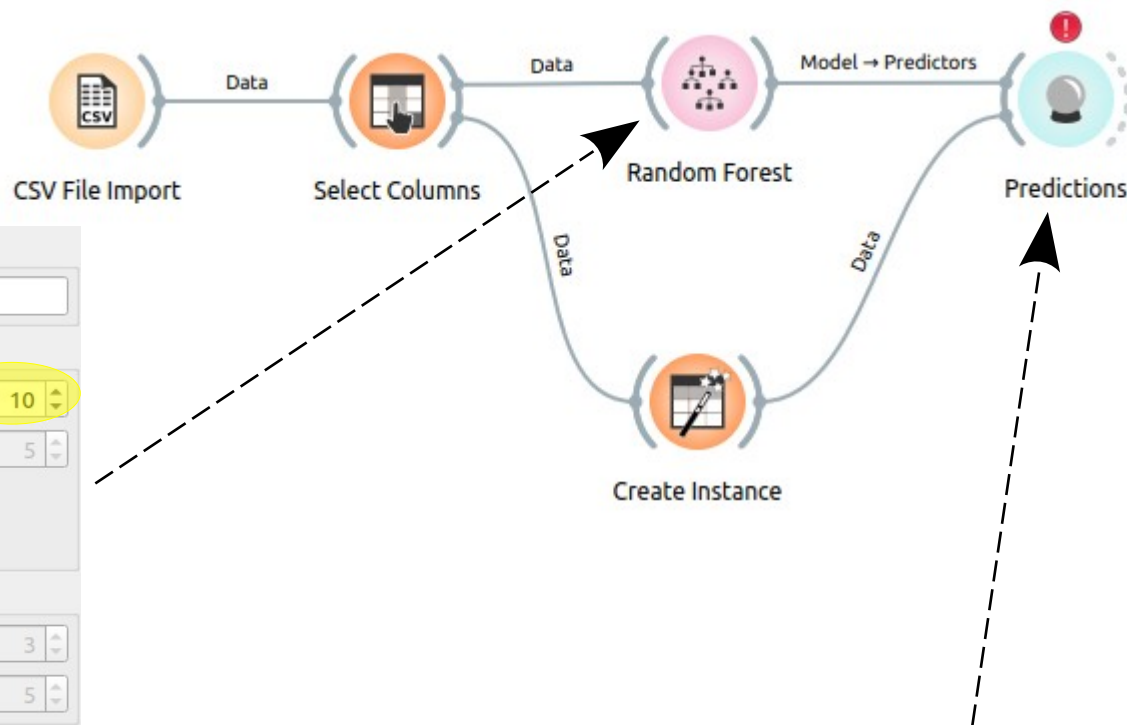
Show probabilities for		Classes in data				<input checked="" type="checkbox"/> Show classification errors	
	Tree	error	geslacht	haarlengte	gewicht	leeftijd	
1	1.00 : 0.00 → M		?	1	180	36	

Bossen

Als je door de bomen het bos niet meer ziet...

- soms berekent men niet 1 boom, maar meerdere
- in plaats van de hoogste gain te nemen, neem een willekeurige kolom (hoogste gain heeft wel meer kans)
- als je nu een beslissing wil nemen: laat alle bomen beslissen en gebruik een “meerderheidsbeslissing”

Bossen



Name
Random Forest

Basic Properties

Number of trees: 10

☐ Number of attributes considered at each split: 5

☐ Replicable training

☐ Balance class distribution

Growth Control

☐ Limit depth of individual trees: 3

☐ Do not split subsets smaller than: 5

Show probabilities for Classes in data ☒ Show classification errors

Random Forest	error	geslacht	haarlengte	gewicht	leeftijd
1 0.90 : 0.10 → M		?	1	180	36

Oefeningen



Oefeningen

- beslissingsbomen
 - play ball
 - scores
 - bank
 - GPT