# Data-Science 1

## association rules


Karel de Grote Hogeschool

1/05/24

## Inhoud

- voorbeelden
- kansen, support en confidence
- apriori algoritme
- FP-Growth algoritme

# Voorbeelden

# Voorbeeld: Spotify

- gegeven: playlists van alle gebruikers in Spotify (sparse matrix)

| | song1 | song2 | song3 | song4 | song5 | song6 | song7 | ... | song 40000000 |
|---|---|---|---|---|---|---|---|---|---|
| playlist1 | | | 1 | | | | | | |
| playlist2 | | | | | | | 1 | | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| playlist 70000000 | | | | 1 | | | | | |

- gebruiker speelt song1355353. Welke liedjes zou deze gebruiker waarschijnlijk graag horen?
- maw welke liedjes hebben de meeste kans om mee in een playlist te zitten met dit liedje?

# Kleiner voorbeeld: winkel

- winkel verkoopt 4 producten: Printer, Papier, Cartridge, Balpen
- er zijn 10 klanten geweest tot nu toe

# Kleiner voorbeeld: winkel

| | Balpen | Cartridge | Papier | Printer |
|---|---|---|---|---|
| 1000123 | TRUE | TRUE | FALSE | TRUE |
| 1000124 | FALSE | TRUE | TRUE | FALSE |
| 1000125 | FALSE | TRUE | TRUE | FALSE |
| 1000126 | FALSE | TRUE | FALSE | TRUE |
| 1000127 | FALSE | TRUE | FALSE | TRUE |
| 1000128 | TRUE | FALSE | FALSE | FALSE |
| 1000129 | TRUE | TRUE | TRUE | FALSE |
| 1000130 | FALSE | TRUE | FALSE | FALSE |
| 1000131 | TRUE | FALSE | FALSE | TRUE |
| 1000132 | FALSE | FALSE | TRUE | FALSE |

# Kleiner voorbeeld: winkel

| | KassaTicket | Product |
|---|---|---|
| 1 | 1000123 | Printer |
| 2 | 1000123 | Cartridge |
| 3 | 1000123 | Balpen |
| 4 | 1000124 | Papier |
| 5 | 1000124 | Cartridge |
| 6 | 1000125 | Papier |
| 7 | 1000125 | Cartridge |
| 8 | 1000126 | Printer |
| 9 | 1000126 | Cartridge |
| 10 | 1000127 | Printer |
| 11 | 1000127 | Cartridge |
| 12 | 1000128 | Balpen |
| 13 | 1000129 | Papier |
| 14 | 1000129 | Cartridge |
| 15 | 1000129 | Balpen |
| 16 | 1000130 | Cartridge |
| 17 | 1000131 | Printer |
| 18 | 1000131 | Balpen |
| 19 | 1000132 | Papier |

# Kleiner voorbeeld: winkel

- gevraagd: als ik weet dat een klant ... koopt, wat is dan de kans dat die ... koopt?

  -> association rules

- dit is een vorm van “unsupervised learning”

# Kansen, support en confidence

2

# Kansen

- notatie
  - P(A) = kans dat A waar is
  - P(B | A) = kans dat B waar is, gegeven dat A waar is
  - kansen zijn getallen tussen 0 en 1 (0% en 100%)
- eigenschap
  - P(A en B) = P(A) * P(B | A)
  - dus: P(B | A) = P(A en B) / P(A)

# Toegepast

- wat is de kans dat iemand een cartridge koopt als je weet dat deze een printer kocht?
  - B = klant koopt cartridge
  - A = klant koopt printer
- P(B | A) = P(A en B) / P(A)
  - P(A en B)
    - ~ (aantal transacties met cartridge en printer)/(aantal transacties)
    - = relatieve frequentie van (cartridge en printer)
    - = 3/10 = 0,3
    - = "**support**" van A en B
  - P(A)
    - ~ (aantal transacties met printer) /(aantal transacties)
    - = relatieve frequentie van printer
    - = 4/10 = 0,4
    - = "**support**" van A
  - dus...
  - men noemt dit de "**confidence**" van deze rule
  - men schrijft de rule ook als: "{printer} => {cartridge}"

| KassaTicket | Printer | Papier | Cartridge | Balpen |
|---|---|---|---|---|
| 1000123 | 1 | 0 | 1 | 1 |
| 1000124 | 0 | 1 | 1 | 0 |
| 1000125 | 0 | 1 | 1 | 0 |
| 1000126 | 1 | 0 | 1 | 0 |
| 1000127 | 1 | 0 | 1 | 0 |
| 1000128 | 0 | 0 | 0 | 1 |
| 1000129 | 0 | 1 | 1 | 1 |
| 1000130 | 0 | 0 | 1 | 0 |
| 1000131 | 1 | 0 | 0 | 1 |
| 1000132 | 0 | 1 | 0 | 0 |

# Het kan ook ingewikkelder

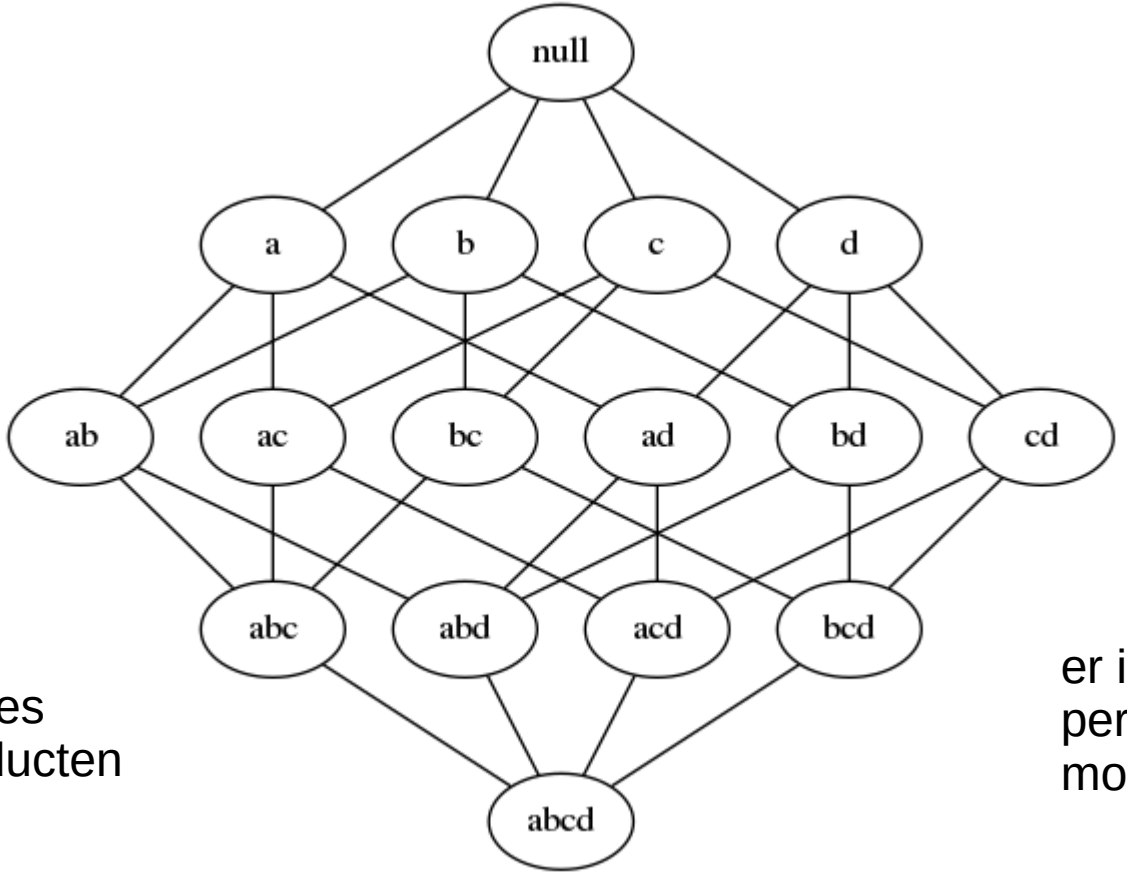
- {Printer, Cartridge} => {Papier}
- P(Papier | Printer en Cartridge)

$$= \frac{P(Papier\ en\ Printer\ en\ Cartridge)}{P(Printer\ en\ Cartridge)}$$

- hoeveel van deze rules kan je bedenken?

# Andere combinaties

a = balpen
b = cartridge
c = papier
d = printer




er zijn 2^n nodes
n = aantal producten

er is een mogelijke rule per lijn: n*2^(n-1) mogelijkheden

# Alle support berekenen

support("balpen") = 0,4
support("cartridge") = 0,7
support("papier") = 0,4
support("printer") = 0,4

support("balpen" en "cartridge") = 0,2
support("balpen" en "papier") = 0,1
support("balpen" en "printer") = 0,2
support("cartridge" en "papier") = 0,3
support("cartridge" en "printer") = 0,3
support("papier" en "printer") = 0

support("balpen" en "cartridge" en "papier") = 0,1
support("balpen" en "cartridge" en "printer") = 0,1
support("cartridge" en "papier" en "printer") = 0
support("balpen" en "papier" en "printer") = 0
support("balpen" en "papier" en "cartridge" en "printer") = 0

# Alle confidence berekenen

- mogelijke regels
  - {balpen} => {cartridge}
  - {balpen} => {papier}
  - {balpen} => {printer}
  - {cartridge} => {balpen}
  - {cartridge} => {papier}
  - {cartridge} => {printer}
  - ...
  - {balpen, cartridge} => {papier}
  - {balpen, cartridge} => {printer}
  - ...
  - {balpen, cartridge, papier} => {printer}
  - ...

- als n=4 --> 32 regels
- als n=10 --> 5 120 regels
- als n=20 --> 10 485 760 regels
- als n=30 --> 16 106 127 360 regels
- als n=100 --> $6{,}338253001 \times 10^{31}$ regels...

# Apriori algoritme

# Limiteren van combinaties

- we zijn enkel geïnteresseerd in rules met hoge confidence

  confidence > min. confidence

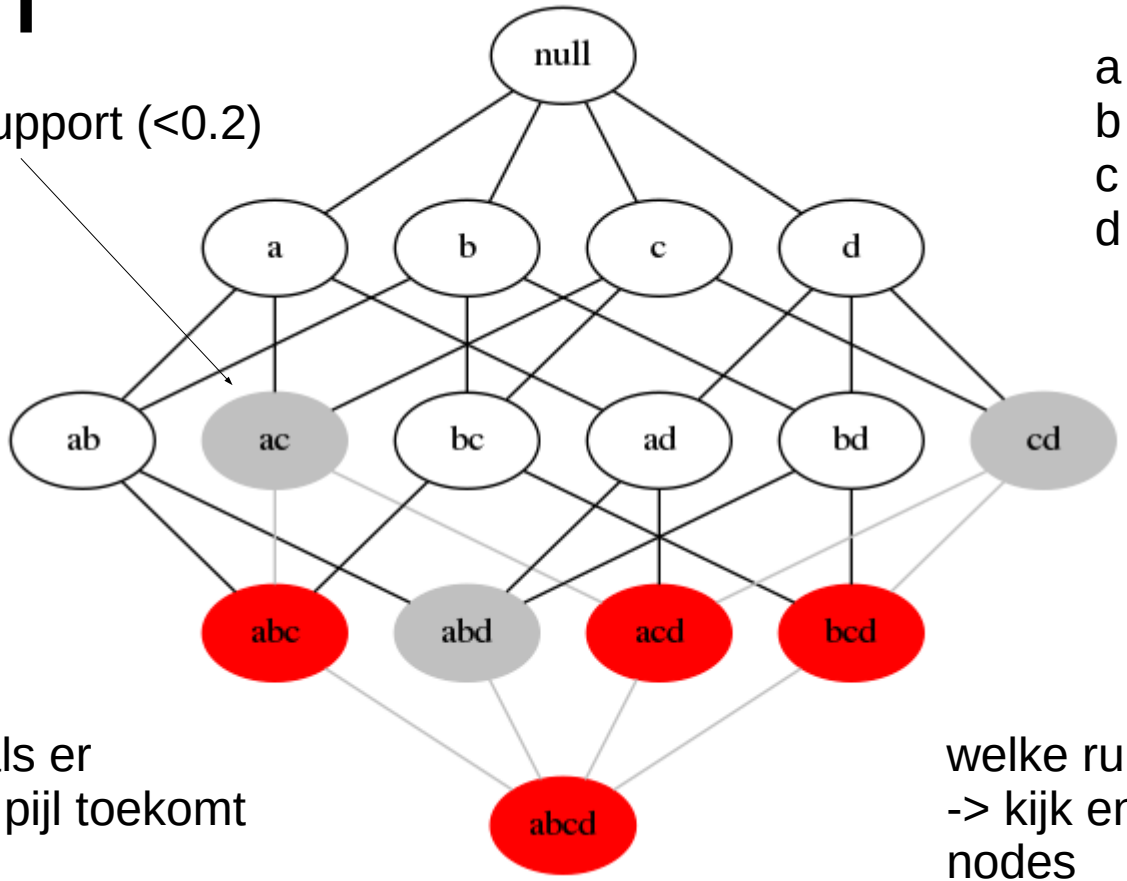
- min. confidence is afhankelijk van de situatie
  - in ons voorbeeld: kies 0,5 (meestal kies je een lagere waarde)
- confidence is hoog als support van (A en B) hoog is
  - bereken dus eerst alle supports
  - ook veel werk

# Berekenen van support

- als de support van A of B klein is, dan is de support van (A en B) ook klein
- dus: breek de berekeningen af als de support te klein wordt
  - min support is ook afhankelijk van de situatie
  - in ons voorbeeld: kies 0,2
- limiteert het aantal mogelijkheden

# Apriori

lage support (<0.2)




a = balpen
b = cartridge
c = papier
d = printer

stop berekenen als er minstens 1 grijze pijl toekomt

welke rules zijn er nog over?
-> kijk enkel nog naar witte nodes

# Resultaten

```
    lhs          rhs          support confidence
[1] {}        => {Cartridge} 0.7     0.70
[2] {Papier}  => {Cartridge} 0.3     0.75
[3] {Printer} => {Balpen}    0.2     0.50
[4] {Balpen}  => {Printer}   0.2     0.50
[5] {Printer} => {Cartridge} 0.3     0.75
[6] {Balpen}  => {Cartridge} 0.2     0.50
```

# Resultaten

```
    lhs           rhs          support confidence
[1] {}         => {Cartridge} 0.7     0.70
[2] {Papier}   => {Cartridge} 0.3     0.75
[3] {Printer}  => {Balpen}    0.2     0.50
[4] {Balpen}   => {Printer}   0.2     0.50
[5] {Printer}  => {Cartridge} 0.3     0.75
[6] {Balpen}   => {Cartridge} 0.2     0.50
```

kijk naar {Balpen} => {Cartridge}

- confidence=0.5
- confidence({} =>Cartridge) = 0.7
- het kopen van een balpen heeft dus een negatieve invloed op het kopen van een cartridge!

# Resultaten

```
    lhs           rhs           support confidence lift
[1] {}         => {Cartridge} 0.7     0.70       1.0000000
[2] {Papier}   => {Cartridge} 0.3     0.75       1.0714286
[3] {Printer}  => {Balpen}    0.2     0.50       1.2500000
[4] {Balpen}   => {Printer}   0.2     0.50       1.2500000
[5] {Printer}  => {Cartridge} 0.3     0.75       1.0714286
[6] {Balpen}   => {Cartridge} 0.2     0.50       0.7142857
```

kijk naar {Balpen} => {Cartridge}

- confidence=0.5
- confidence({} =>Cartridge) = 0.7
- het kopen van een balpen heeft dus een negatieve invloed op het kopen van een cartridge!
- je kan dit zien aan de **lift** (dit is P(A en B)/P(A)/P(B))
- als lift < 1 wordt de regel veelal geschrapt

# FP Growth algoritme

4

# Nadeel apriori

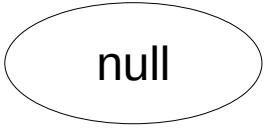
- je moet alle transacties steeds terug doorlopen
- dit vraagt nog steeds heel veel tijd
- als de data in een ander formaat staat, kan dat sneller

# FP Growth

- zet de data om naar een boomstructuur
  - deze neemt veel minder plaats in beslag dan de sparse matrix
- algoritme
  - bereken de supports van alle producten
  - schrap alle supports < min. support
  - sorteer de producten van grote naar kleine support
  - maak 1 node (null) voor de boom
  - per transactie
    - sorteer de producten in de transactie volgens vorige sortering
    - voeg de transactie in de boom in, maak nodes bij indien nodig
    - verhoog teller bij iedere node die je hergebruikt
    - verbind nodes die hetzelfde product bevatten met een aparte lijn

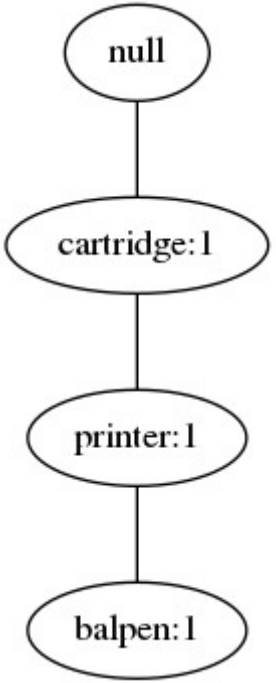
# FP Growth voorbeeld

- sortering = Cartridge (0,7), Printer (0,4), Papier (0,4), Balpen (0,4)
- initiële boom is



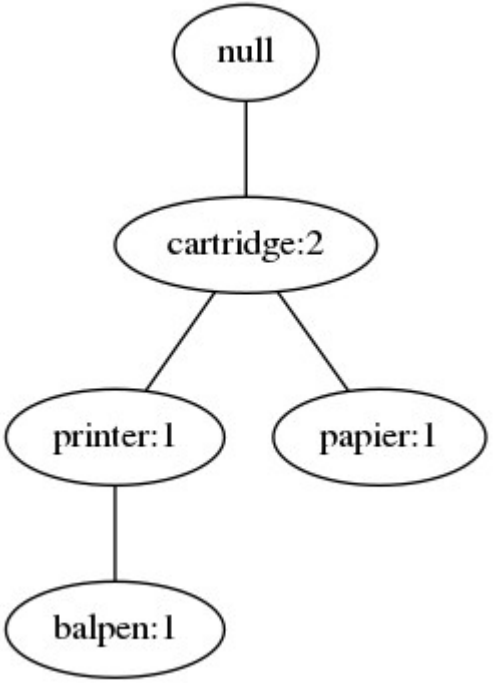
# FP Growth voorbeeld

- transactie 1: Cartridge, Printer, Balpen
- resultaat:



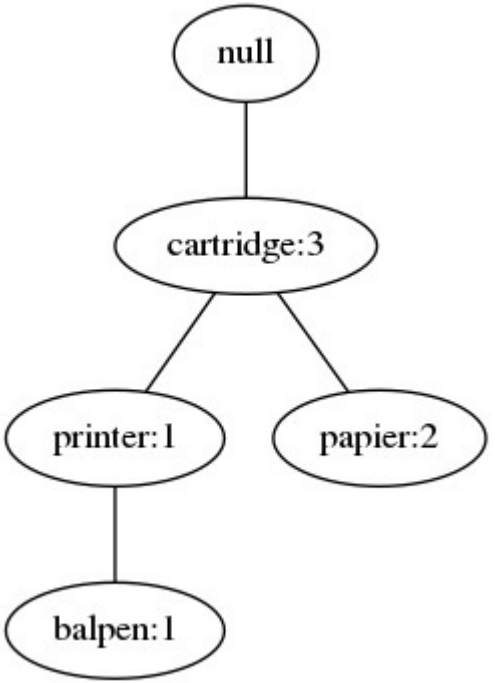
# FP Growth voorbeeld

- transactie 2: Cartridge, Papier
- resultaat:



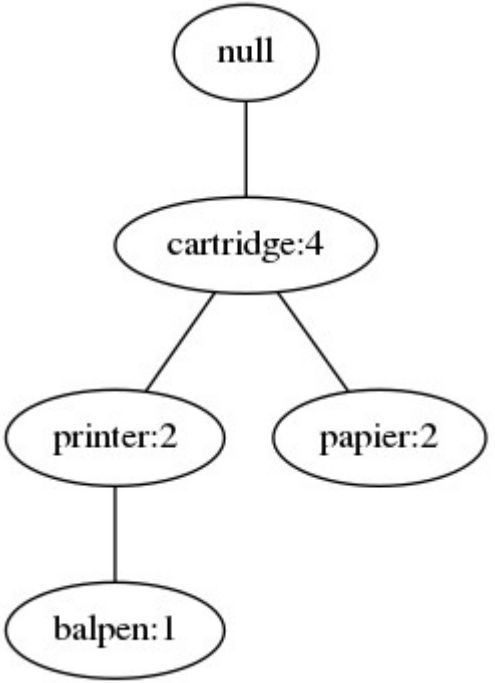
# FP Growth voorbeeld

- transactie 3: Cartridge, Papier
- resultaat:



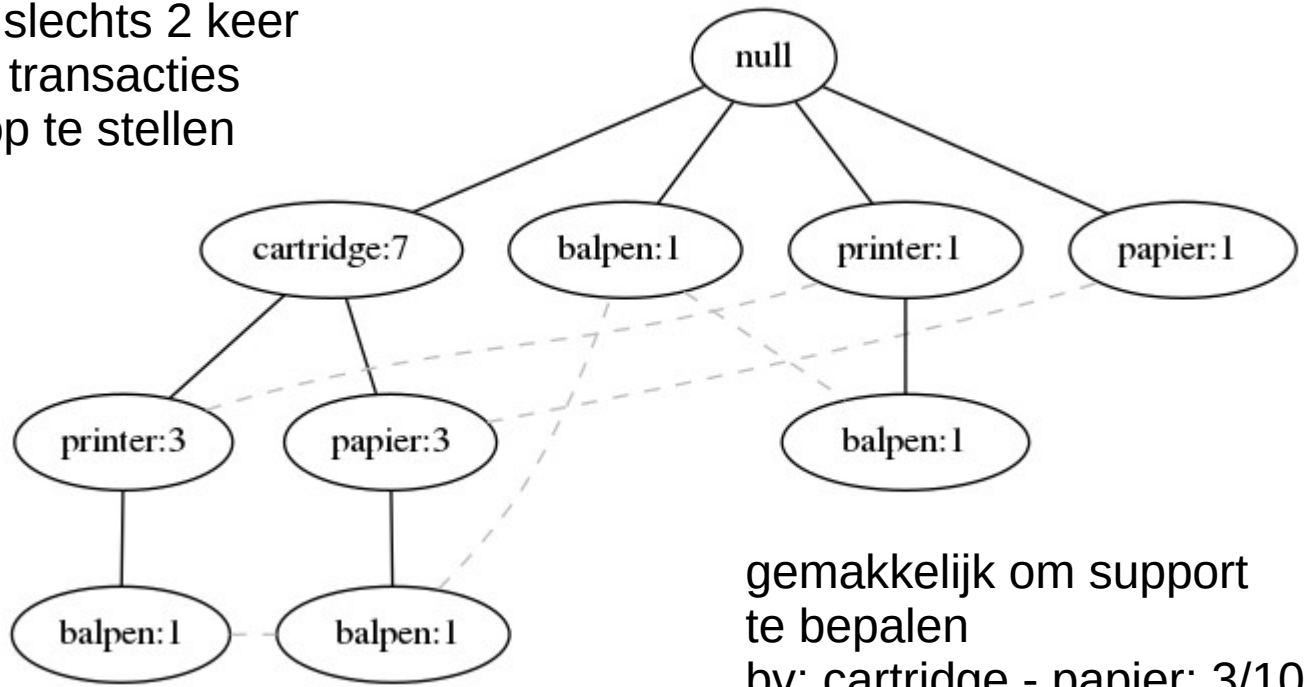
# FP Growth voorbeeld

- transactie 4: Cartridge, Printer
- resultaat:



# FP Growth resultaat

je moet slechts 2 keer
door de transacties
om dit op te stellen

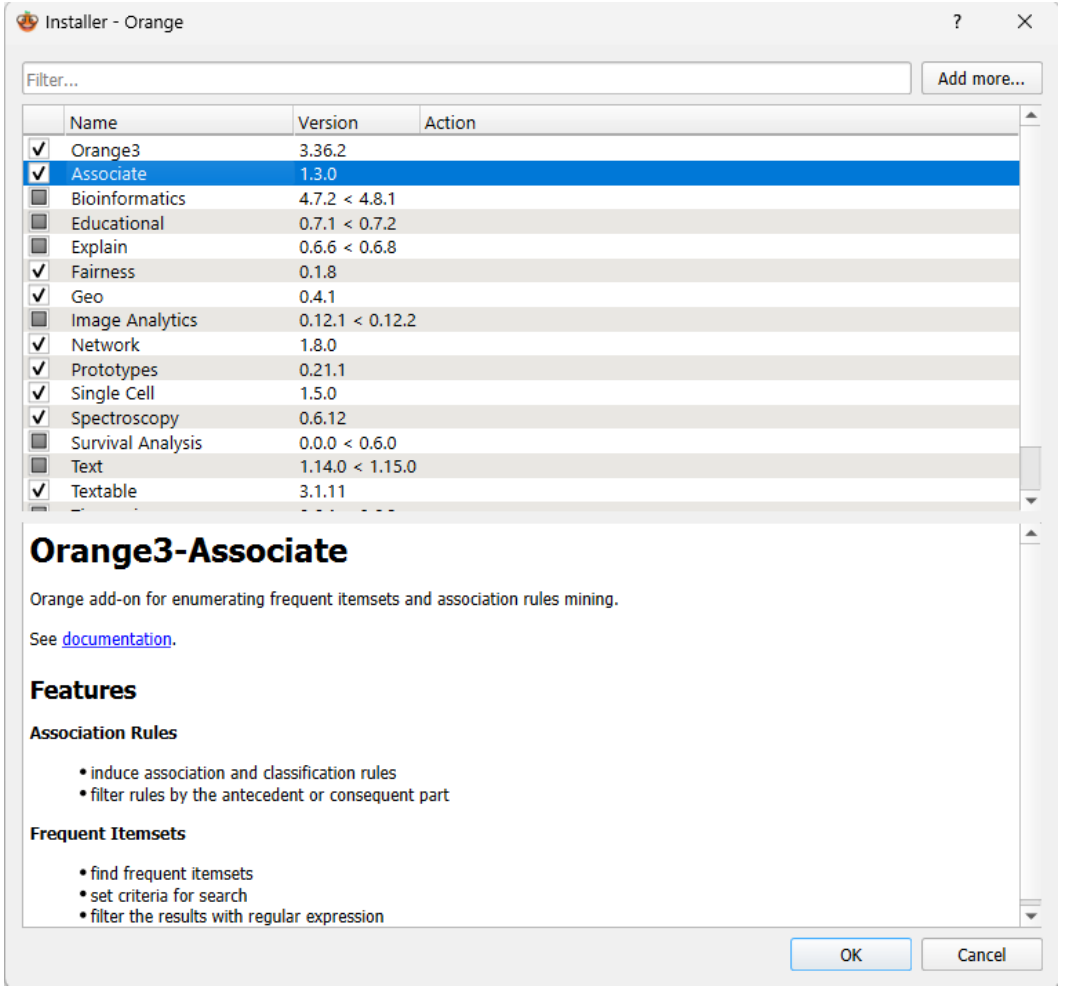



gemakkelijk om support
te bepalen
bv: cartridge - papier: 3/10
bv: printer-balpen: (1+1)/10 (volg de links)


Orange

# Package

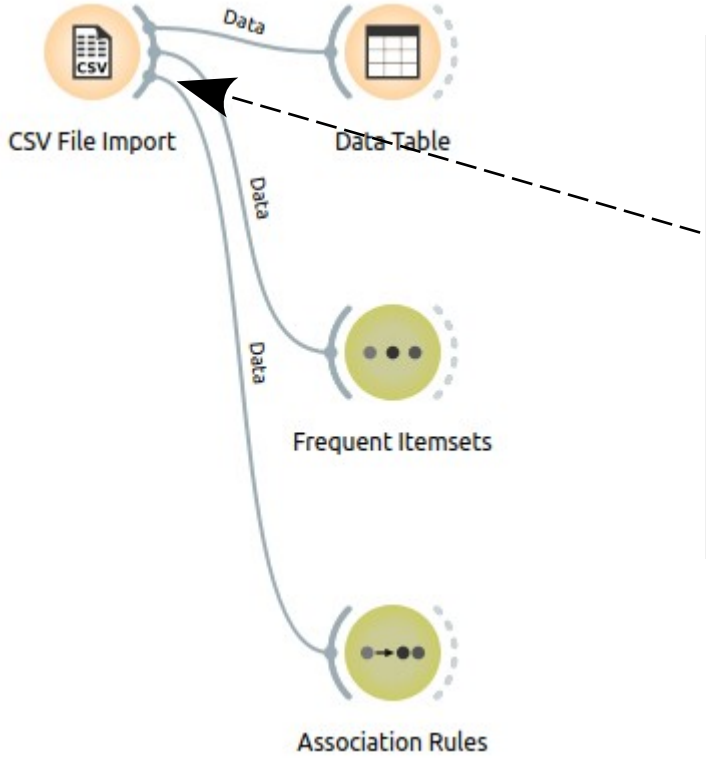
- installeer Associate package



# Aandachtspunten

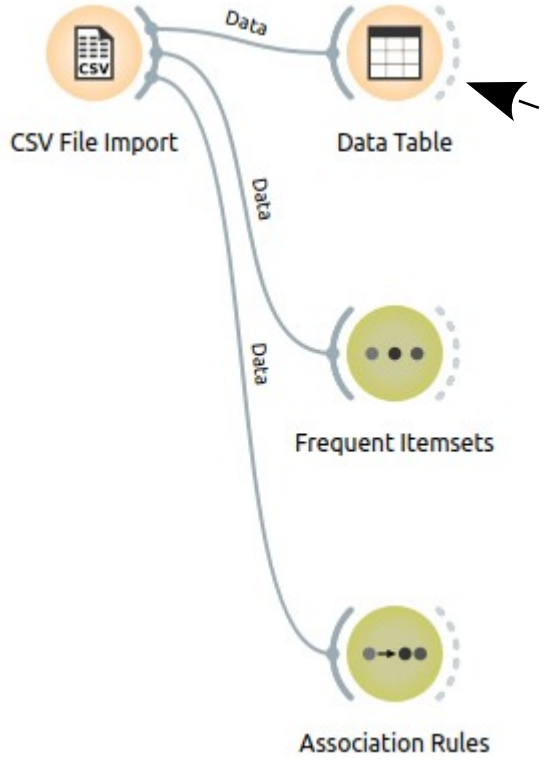
- formaat moet juist zijn
  - sparse matrix
  - enkel dingen aanduiden die gekocht zijn
  - geen waarden (? of NA) voor dingen die niet gekocht zijn
- fout formaat leidt tot foute regels
  - soms moet je pre-processen
- opletten voor transactie ids
  - mogen niet opgenomen worden in resultaten
  - best zelf verwijderen met skip of Select Columns widget

# Voorbeeld



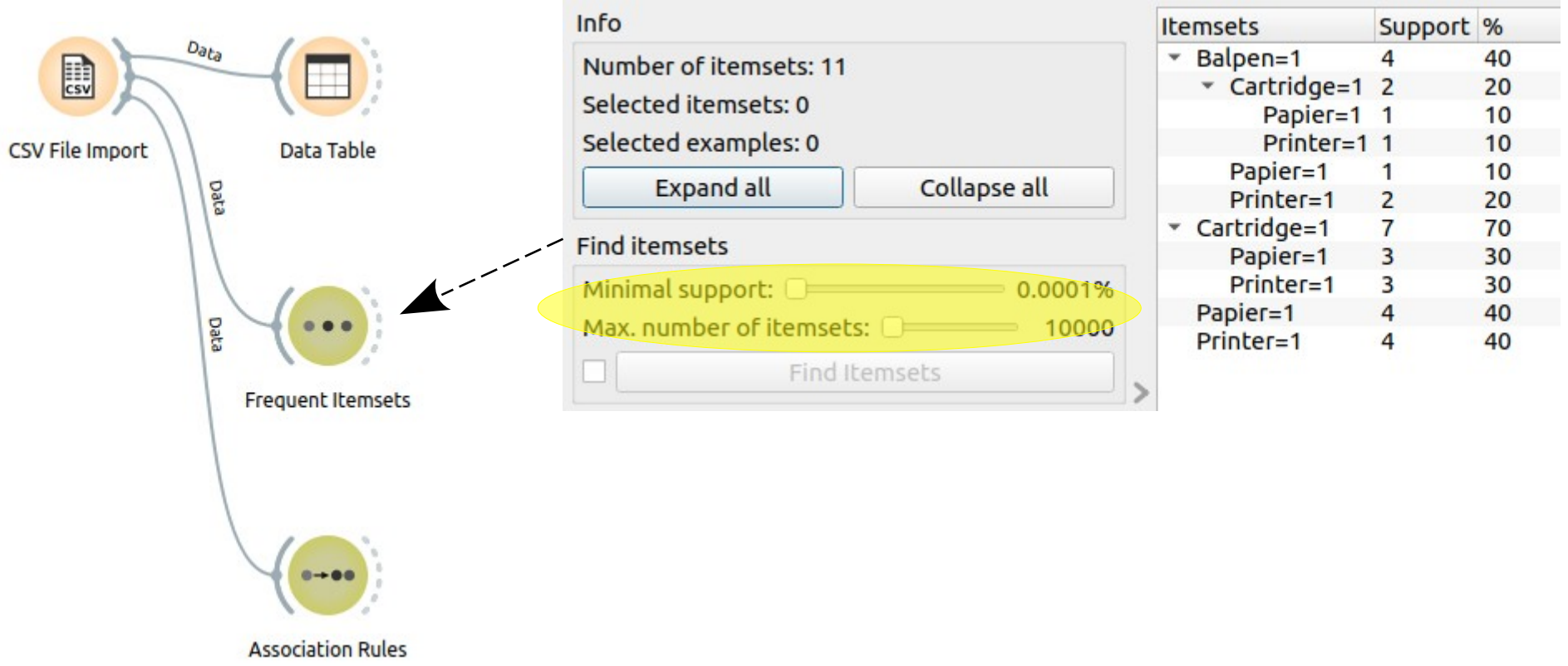

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | KassaTicket | Balpen | Cartridge | Papier | Printer |
| 2 | 1000123 | 1 | 1 | | 1 |
| 3 | 1000124 | | 1 | 1 | |
| 4 | 1000125 | | 1 | 1 | |
| 5 | 1000126 | | 1 | | 1 |
| 6 | 1000127 | | 1 | | 1 |
| 7 | 1000128 | 1 | | | |
| 8 | 1000129 | 1 | 1 | 1 | |
| 9 | 1000130 | | 1 | | |
| 10 | 1000131 | 1 | | | 1 |
| 11 | 1000132 | | | 1 | |

# Voorbeeld

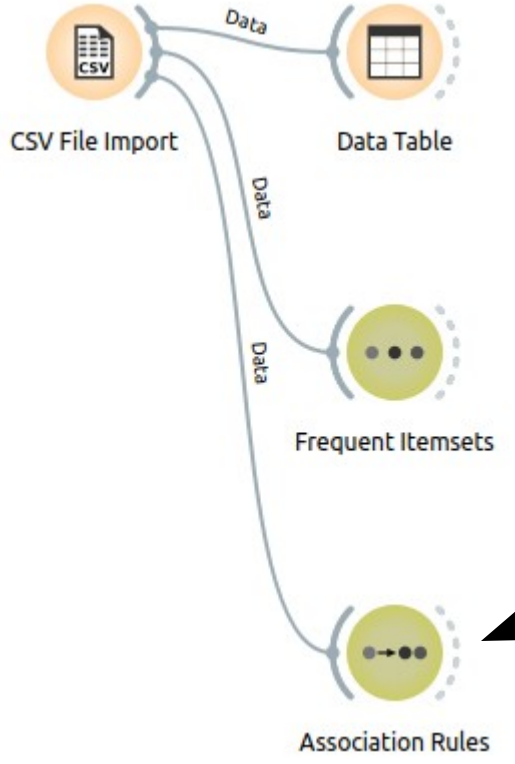



| | Balpen | Cartridge | Papier | Printer |
|---|---|---|---|---|
| 1 | 1 | 1 | ? | 1 |
| 2 | ? | 1 | 1 | ? |
| 3 | ? | 1 | 1 | ? |
| 4 | ? | 1 | ? | 1 |
| 5 | ? | 1 | ? | 1 |
| 6 | 1 | ? | ? | ? |
| 7 | 1 | 1 | 1 | ? |
| 8 | ? | 1 | ? | ? |
| 9 | 1 | ? | ? | 1 |
| 10 | ? | ? | 1 | ? |

# Voorbeeld


Data
CSV File Import
Data Table
Data
Data
Frequent Itemsets
Association Rules
Info
Number of itemsets: 11
Selected itemsets: 0
Selected examples: 0
Expand all
Collapse all
Find itemsets
Minimal support: 0.0001%
Max. number of itemsets: 10000
Find Itemsets
Itemsets Support %
Balpen=1 4 40
Cartridge=1 2 20
Papier=1 1 10
Printer=1 1 10
Papier=1 1 10
Printer=1 2 20
Cartridge=1 7 70
Papier=1 3 30
Printer=1 3 30
Papier=1 4 40
Printer=1 4 40

# Voorbeeld




Info

Rules: 22 (shown 22)

Find association rules

Min. supp.: 0.0001 %

Min. conf.: 1 %

Max. rules: 10k

Induce only classification rules

Restrict search by below filters

Find Rules

Filter by Antecedent

Contains:

Items, min: 1 max: 999

Filter by Consequent

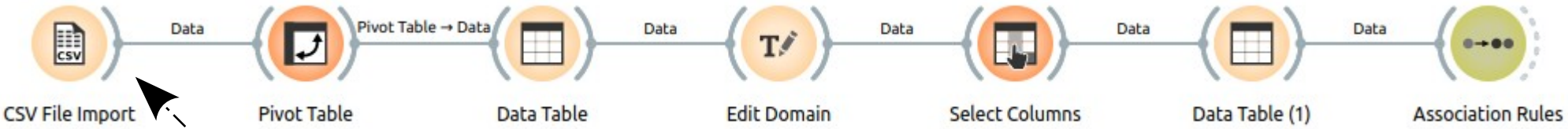
Contains:

Items, min: 1 max: 999

Send selection

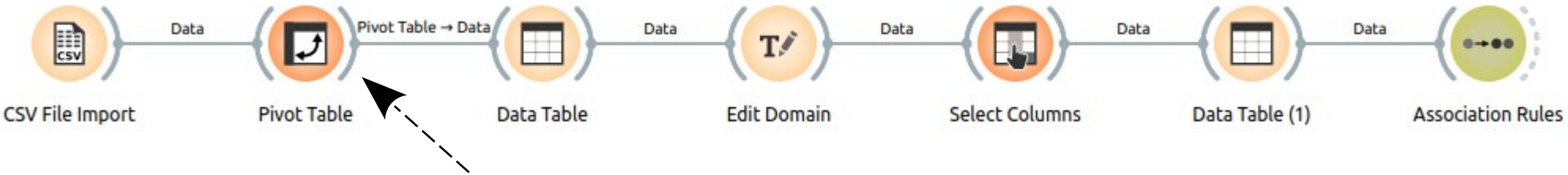
| Supp | Conf | Covr | Strg | Lift | Levr | Antecedent | | Consequent |
|---|---|---|---|---|---|---|---|---|
| 0.200 | 0.286 | 0.700 | 0.571 | 0.714 | -0.080 | Cartridge=1 | → | Balpen=1 |
| 0.200 | 0.500 | 0.400 | 1.750 | 0.714 | -0.080 | Balpen=1 | → | Cartridge=1 |
| 0.100 | 0.250 | 0.400 | 1.000 | 0.625 | -0.060 | Papier=1 | → | Balpen=1 |
| 0.100 | 0.250 | 0.400 | 1.000 | 0.625 | -0.060 | Balpen=1 | → | Papier=1 |
| 0.300 | 0.750 | 0.400 | 1.750 | 1.071 | 0.020 | Papier=1 | → | Cartridge=1 |
| 0.300 | 0.429 | 0.700 | 0.571 | 1.071 | 0.020 | Cartridge=1 | → | Papier=1 |
| 0.100 | 0.333 | 0.300 | 1.333 | 0.833 | -0.020 | Cartridge=1, Papier=1 | → | Balpen=1 |
| 0.100 | 1.000 | 0.100 | 7.000 | 1.429 | 0.030 | Balpen=1, Papier=1 | → | Cartridge=1 |
| 0.100 | 0.250 | 0.400 | 0.500 | 1.250 | 0.020 | Papier=1 | → | Balpen=1, Cartridge=1 |
| 0.100 | 0.500 | 0.200 | 2.000 | 1.250 | 0.020 | Balpen=1, Cartridge=1 | → | Papier=1 |
| 0.100 | 0.143 | 0.700 | 0.143 | 1.429 | 0.030 | Cartridge=1 | → | Balpen=1, Papier=1 |
| 0.100 | 0.250 | 0.400 | 0.750 | 0.833 | -0.020 | Balpen=1 | → | Cartridge=1, Papier=1 |
| 0.200 | 0.500 | 0.400 | 1.000 | 1.250 | 0.040 | Printer=1 | → | Balpen=1 |
| 0.200 | 0.500 | 0.400 | 1.000 | 1.250 | 0.040 | Balpen=1 | → | Printer=1 |
| 0.300 | 0.750 | 0.400 | 1.750 | 1.071 | 0.020 | Printer=1 | → | Cartridge=1 |
| 0.300 | 0.429 | 0.700 | 0.571 | 1.071 | 0.020 | Cartridge=1 | → | Printer=1 |
| 0.100 | 0.333 | 0.300 | 1.333 | 0.833 | -0.020 | Cartridge=1, Printer=1 | → | Balpen=1 |
| 0.100 | 0.500 | 0.200 | 3.500 | 0.714 | -0.040 | Balpen=1, Printer=1 | → | Cartridge=1 |
| 0.100 | 0.250 | 0.400 | 0.500 | 1.250 | 0.020 | Printer=1 | → | Balpen=1, Cartridge=1 |
| 0.100 | 0.500 | 0.200 | 2.000 | 1.250 | 0.020 | Balpen=1, Cartridge=1 | → | Printer=1 |
| 0.100 | 0.143 | 0.700 | 0.286 | 0.714 | -0.040 | Cartridge=1 | → | Balpen=1, Printer=1 |
| 0.100 | 0.250 | 0.400 | 0.750 | 0.833 | -0.020 | Balpen=1 | → | Cartridge=1, Printer=1 |

# Preprocessing




| | N 1 | C 2 |
|---|---|---|
| 1 | transaction | product |
| 2 | 1 | beer |
| 3 | 1 | fruit |
| 4 | 1 | bread |
| 5 | 2 | bread |
| 6 | 2 | fruit |
| 7 | 2 | car |
| 8 | 3 | beer |
| 9 | 4 | bread |
| 10 | 4 | car |

# Preprocessing

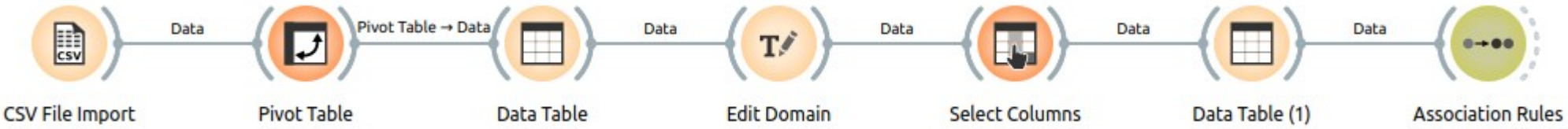



Rows
transaction
Columns
product
Values
(None)
Aggregations
Count

product

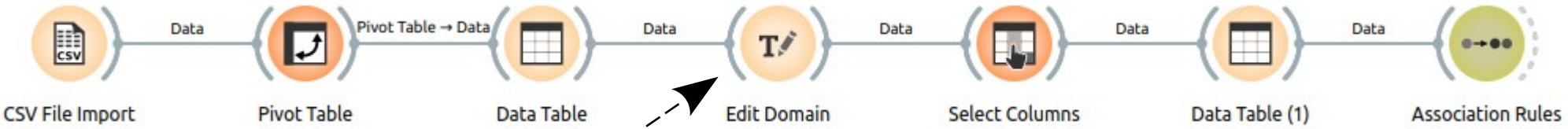
| transaction | Count | beer | bread | car | fruit | Total |
|---|---|---|---|---|---|---|
| | 1 | 1.0 | 1.0 | 0.0 | 1.0 | 3.0 |
| | 2 | 0.0 | 1.0 | 1.0 | 1.0 | 3.0 |
| | 3 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 |
| | 4 | 0.0 | 1.0 | 1.0 | 0.0 | 2.0 |
| | Total | 2.0 | 3.0 | 2.0 | 2.0 | 9.0 |

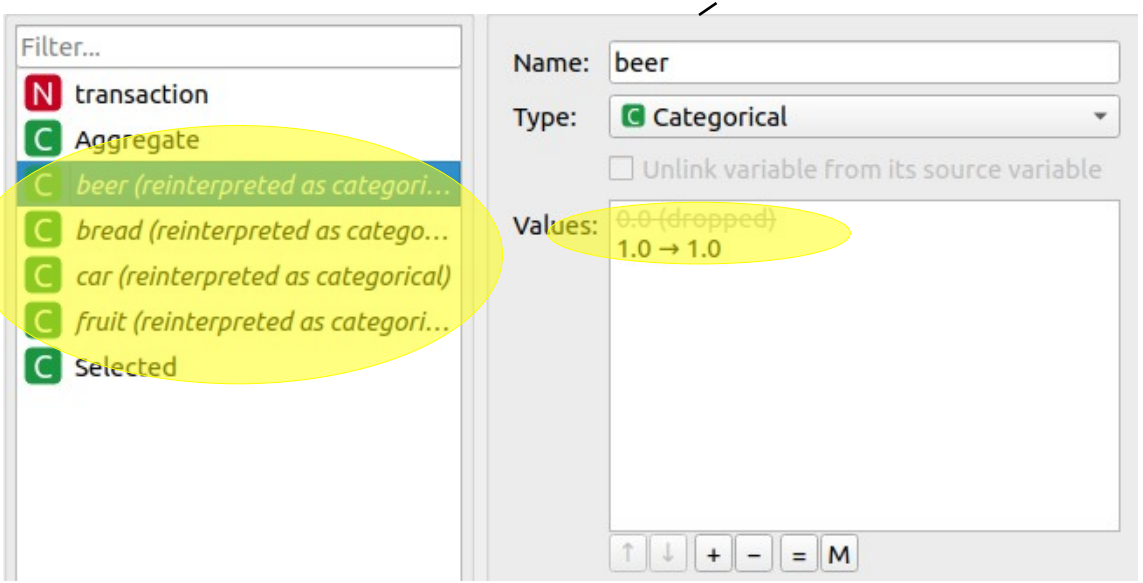
# Preprocessing



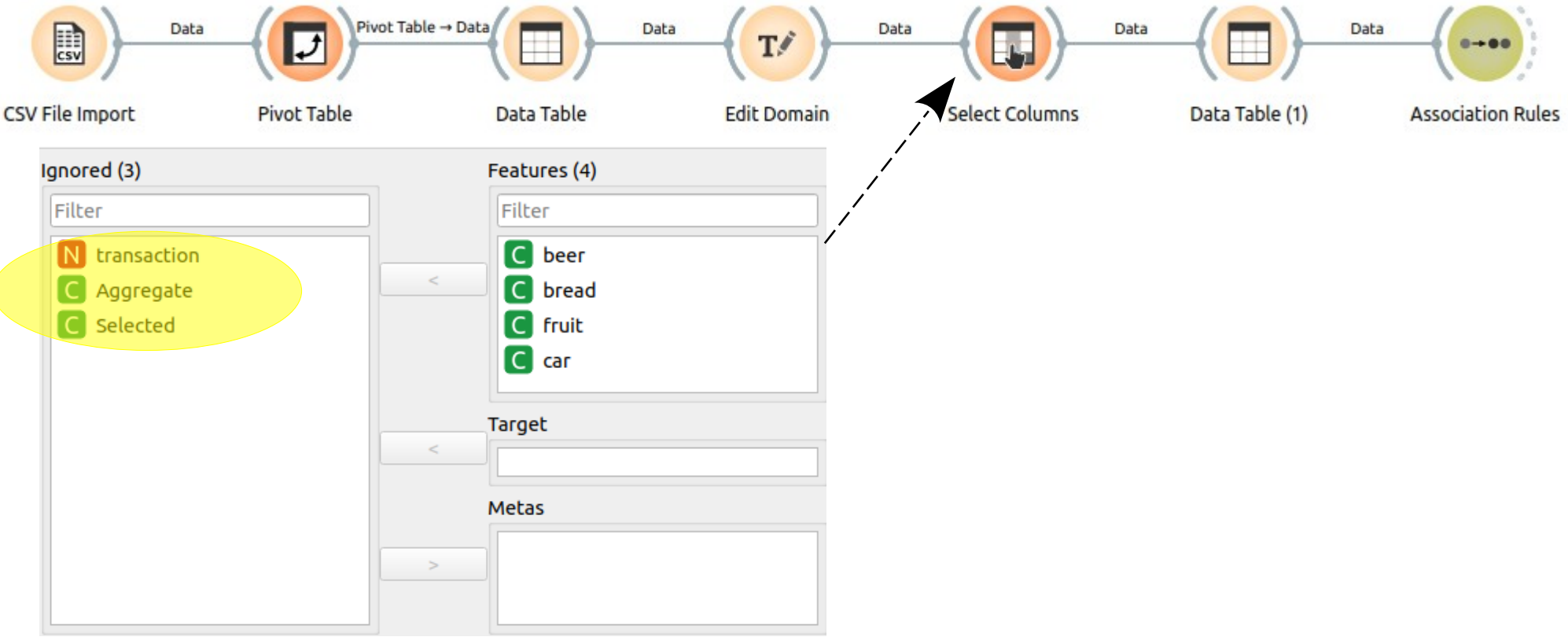

| | transaction | Aggregate | beer | bread | car | fruit |
|---|---|---|---|---|---|---|
| 1 | 1 | Count | 1.0 | 1.0 | 0.0 | 1.0 |
| 2 | 2 | Count | 0.0 | 1.0 | 1.0 | 1.0 |
| 3 | 3 | Count | 1.0 | 0.0 | 0.0 | 0.0 |
| 4 | 4 | Count | 0.0 | 1.0 | 1.0 | 0.0 |

# Preprocessing

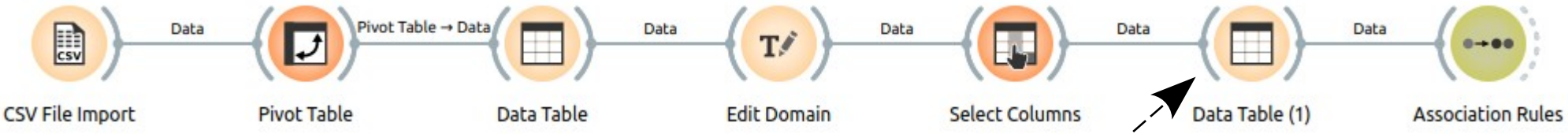


# Preprocessing

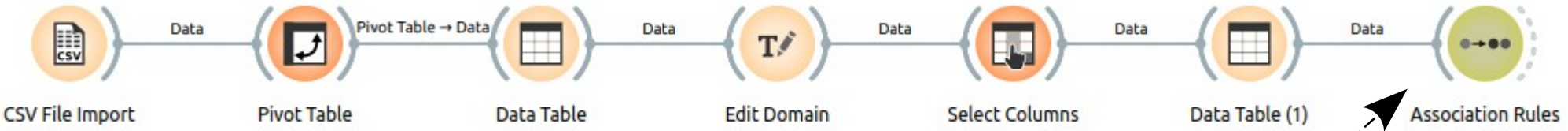


# Preprocessing




| | Selected | beer | bread | fruit | car |
|---|---|---|---|---|---|
| 1 | No | 1.0 | 1.0 | 1.0 | ? |
| 2 | No | ? | 1.0 | 1.0 | 1.0 |
| 3 | No | 1.0 | ? | ? | ? |
| 4 | No | ? | 1.0 | ? | 1.0 |

# Preprocessing




Info

Rules: 19 (shown 19)

Find association rules

Min. supp.: 5 %

Min. conf.: 35 %

Max. rules: 10k

Induce only classification rules

Restrict search by below filters

Find Rules

Filter by Antecedent

Contains:

Items, min: 1 max: 999

Filter by Consequent

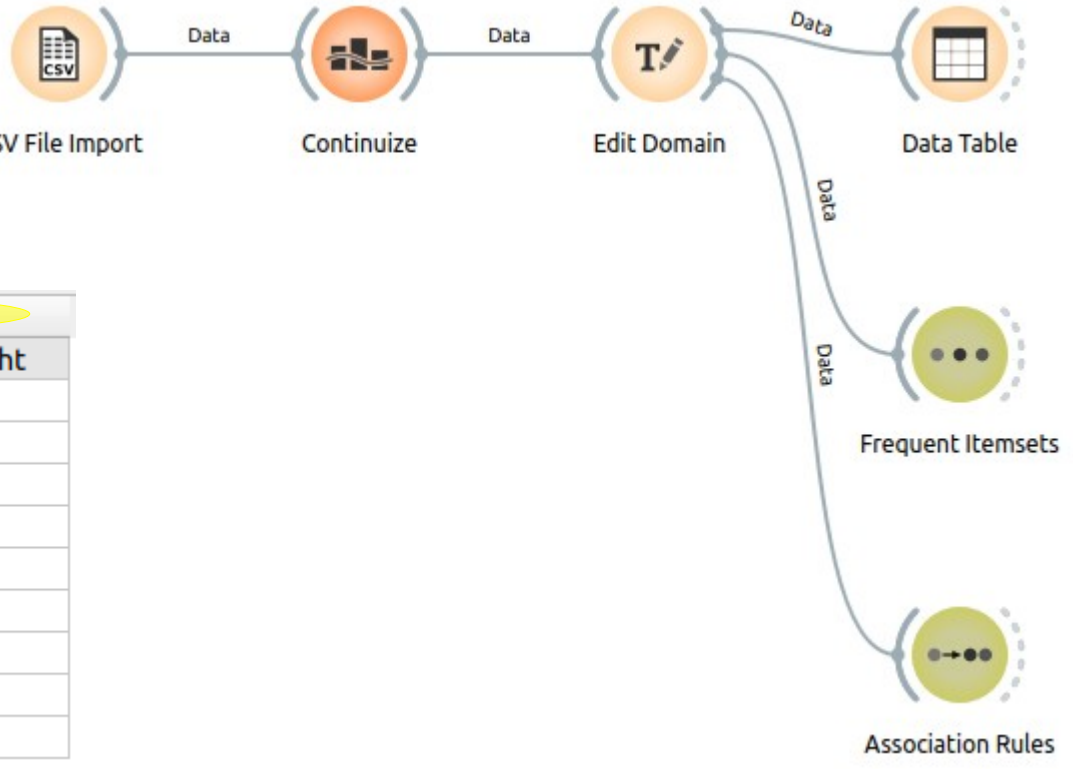
Contains:

Items, min: 1 max: 999

| Supp | Conf | Covr | Strg | Lift | Levr | Antecedent | | |
|---|---|---|---|---|---|---|---|---|
| 0.250 | 0.500 | 0.500 | 1.500 | 0.667 | -0.125 | beer=1.0 | → | bread=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | fruit=1.0 | → | beer=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | beer=1.0 | → | fruit=1.0 |
| 0.500 | 1.000 | 0.500 | 1.500 | 1.333 | 0.125 | fruit=1.0 | → | bread=1.0 |
| 0.500 | 0.667 | 0.750 | 0.667 | 1.333 | 0.125 | bread=1.0 | → | fruit=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | bread=1.0, fruit=1.0 | → | beer=1.0 |
| 0.250 | 1.000 | 0.250 | 3.000 | 1.333 | 0.062 | beer=1.0, fruit=1.0 | → | bread=1.0 |
| 0.250 | 0.500 | 0.500 | 0.500 | 2.000 | 0.125 | fruit=1.0 | → | beer=1.0, bread=1.0 |
| 0.250 | 1.000 | 0.250 | 2.000 | 2.000 | 0.125 | beer=1.0, bread=1.0 | → | fruit=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | beer=1.0 | → | bread=1.0, fruit=1.0 |
| 0.500 | 1.000 | 0.500 | 1.500 | 1.333 | 0.125 | car=1.0 | → | bread=1.0 |
| 0.500 | 0.667 | 0.750 | 0.667 | 1.333 | 0.125 | bread=1.0 | → | car=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | car=1.0 | → | fruit=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | fruit=1.0 | → | car=1.0 |
| 0.250 | 1.000 | 0.250 | 3.000 | 1.333 | 0.062 | fruit=1.0, car=1.0 | → | bread=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | bread=1.0, car=1.0 | → | fruit=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | car=1.0 | → | bread=1.0, fruit=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | bread=1.0, fruit=1.0 | → | car=1.0 |
| 0.250 | 0.500 | 0.500 | 1.000 | 1.000 | 0.000 | fruit=1.0 | → | bread=1.0, car=1.0 |

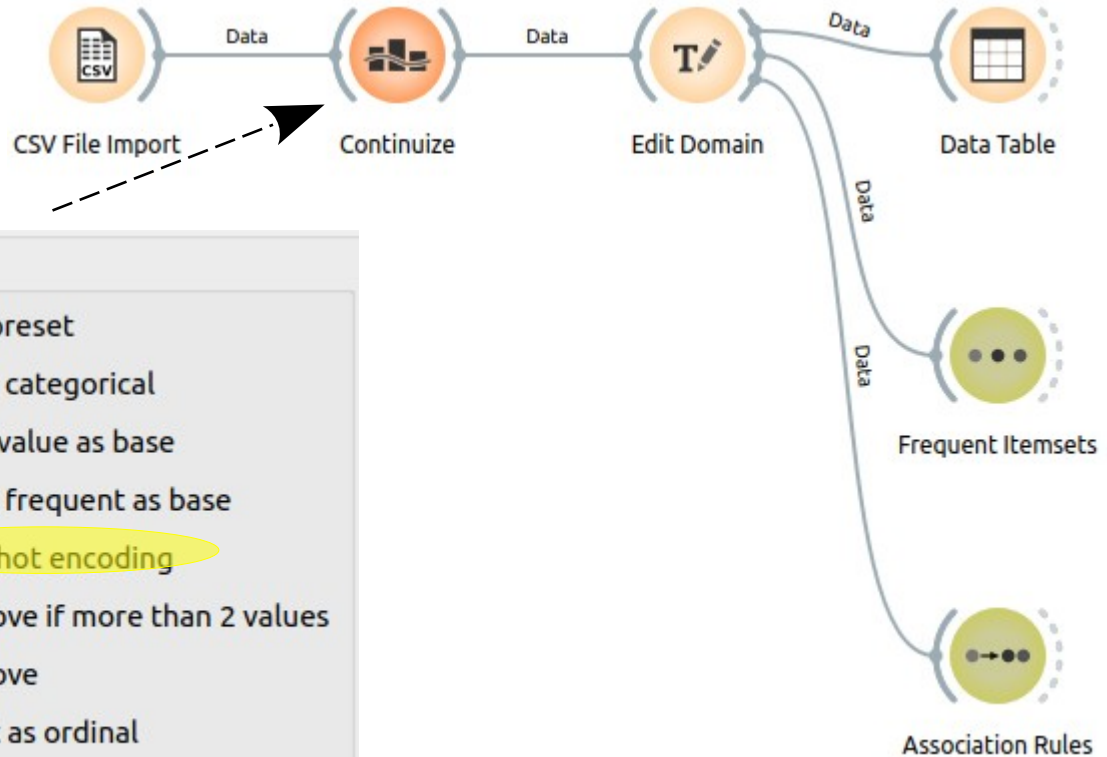
# Voorbeeld: simpsons

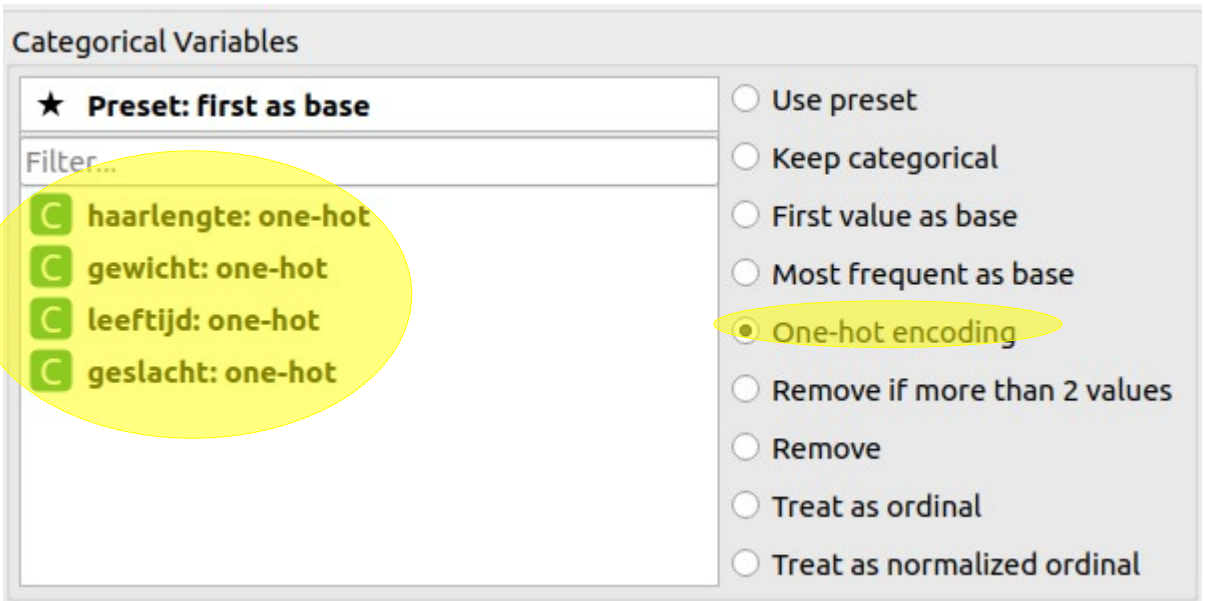
# Voorbeeld simpsons



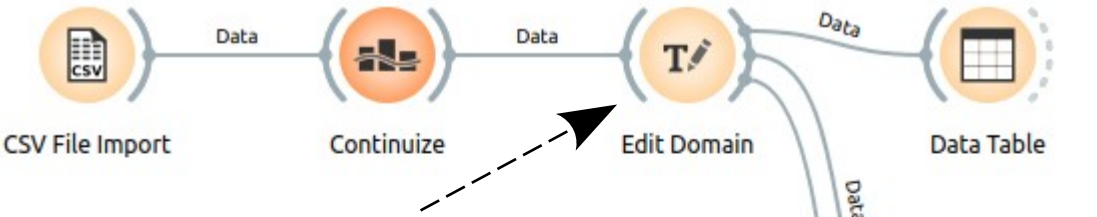

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | haarlengte | gewicht | leeftijd | geslacht |
| 2 | <3 | >160 | 30-40 | M |
| 3 | >5 | <=160 | 30-40 | V |
| 4 | <3 | <=160 | <30 | M |
| 5 | >5 | <=160 | <30 | V |
| 6 | 3-5 | <=160 | <30 | V |
| 7 | <3 | >160 | >40 | M |
| 8 | >5 | <=160 | >40 | V |
|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

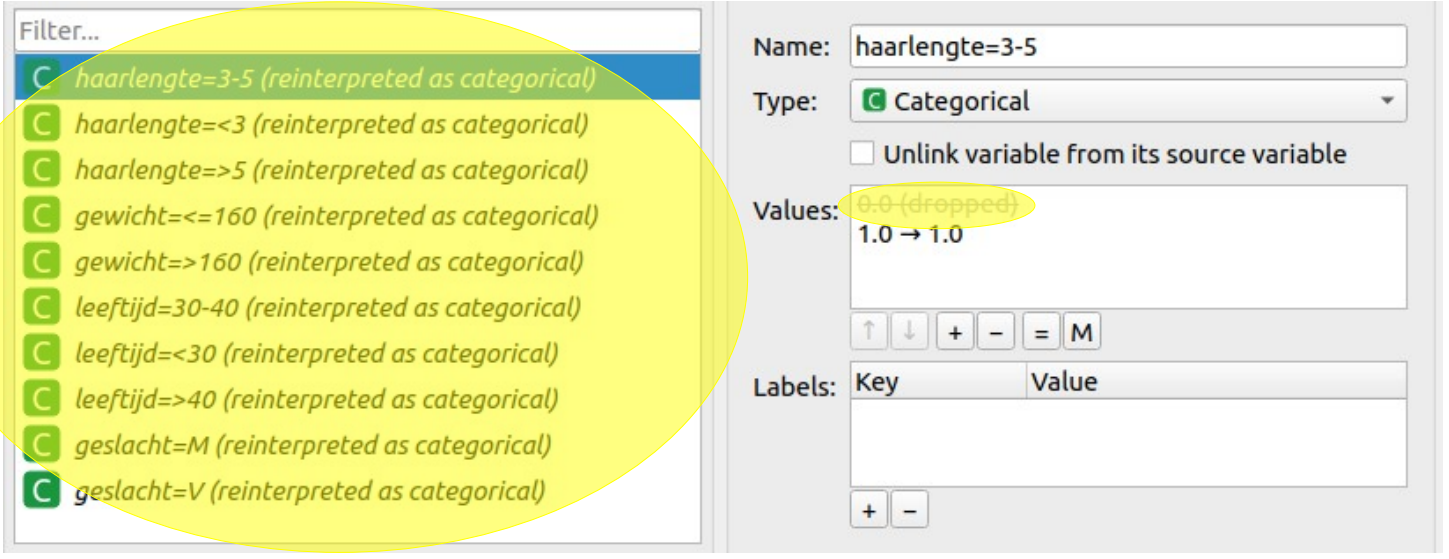
# Voorbeeld simpsons



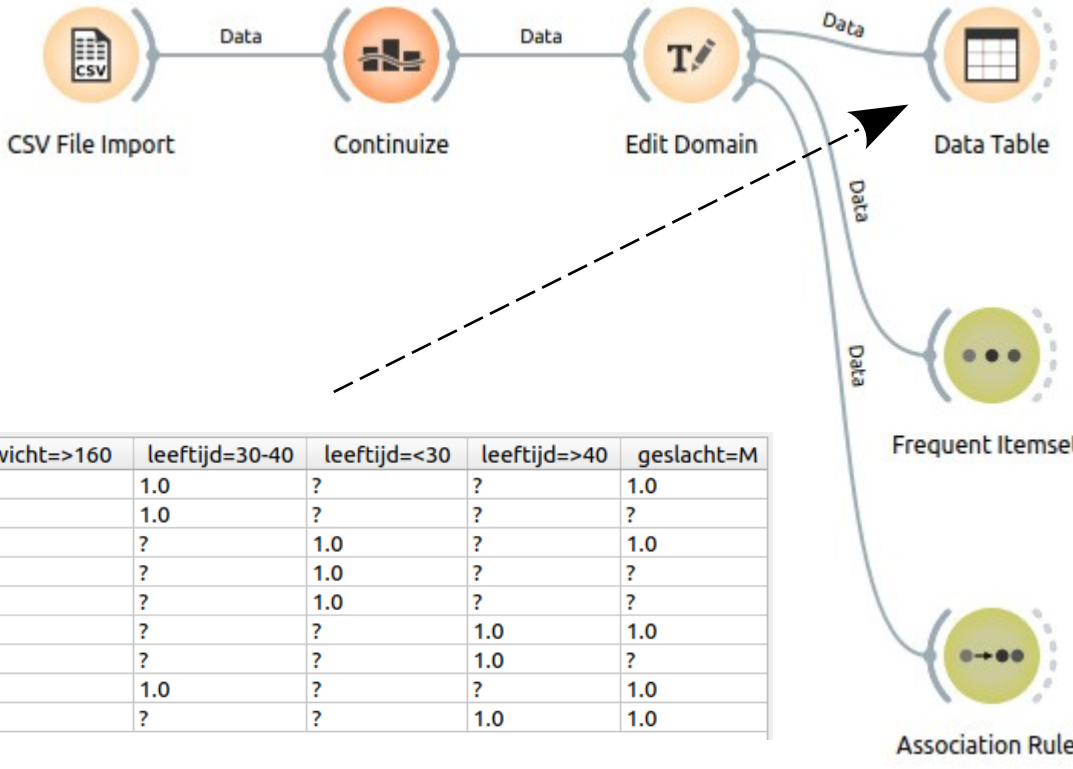



# Voorbeeld simpsons



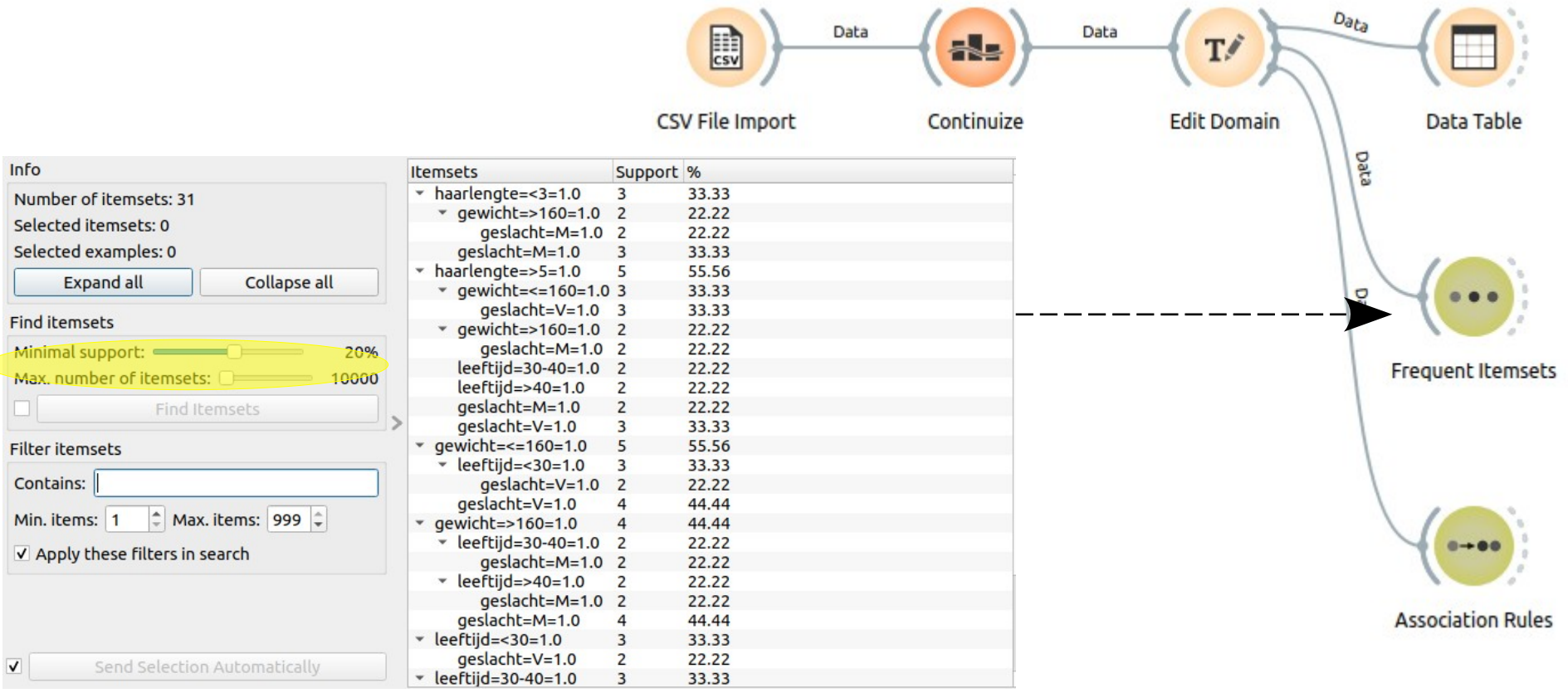



# Voorbeeld simpsons

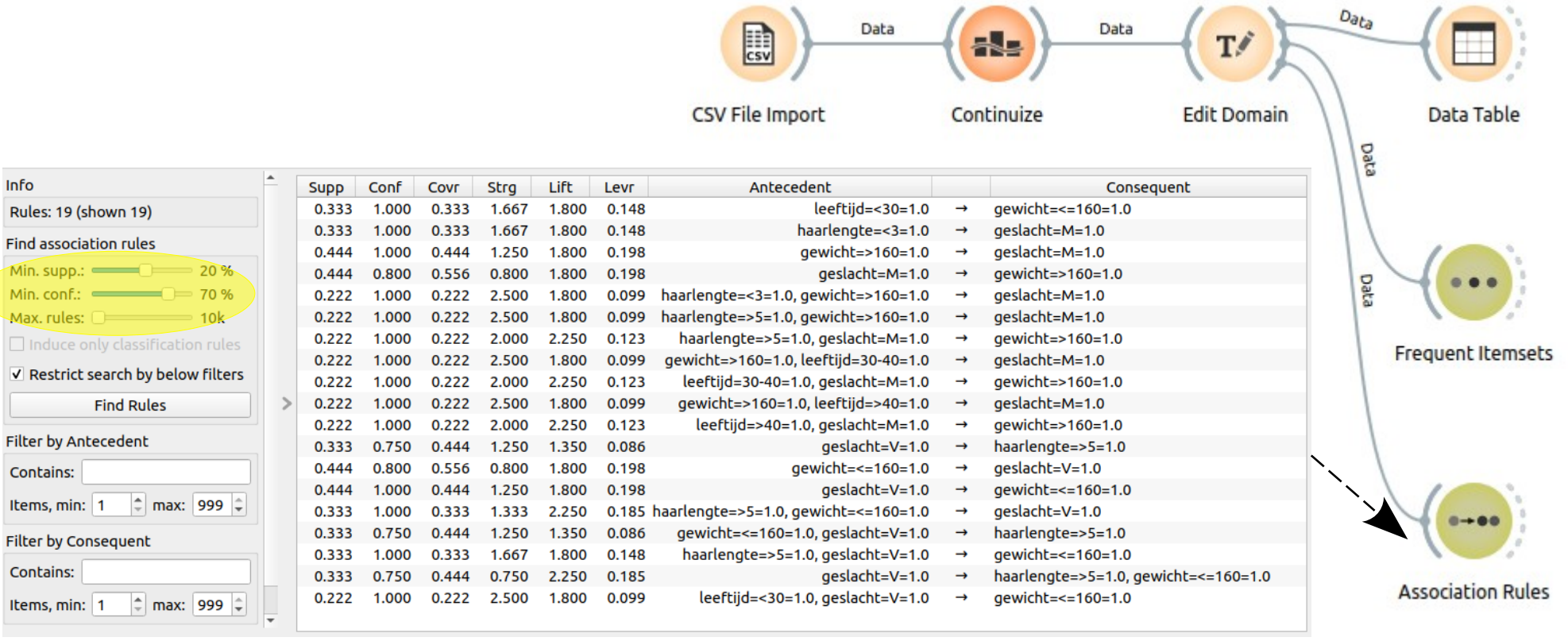



| | haarlengte=3-5 | haarlengte=<3 | haarlengte=>5 | gewicht=<=160 | gewicht=>160 | leeftijd=30-40 | leeftijd=<30 | leeftijd=>40 | geslacht=M |
|---|---|---|---|---|---|---|---|---|---|
| 1 | ? | 1.0 | ? | ? | 1.0 | 1.0 | ? | ? | 1.0 |
| 2 | ? | ? | 1.0 | 1.0 | ? | 1.0 | ? | ? | ? |
| 3 | ? | 1.0 | ? | 1.0 | ? | ? | 1.0 | ? | 1.0 |
| 4 | ? | ? | 1.0 | 1.0 | ? | ? | 1.0 | ? | ? |
| 5 | 1.0 | ? | ? | 1.0 | ? | ? | 1.0 | ? | ? |
| 6 | ? | 1.0 | ? | ? | 1.0 | ? | ? | 1.0 | 1.0 |
| 7 | ? | ? | 1.0 | 1.0 | ? | ? | ? | 1.0 | ? |
| 8 | ? | ? | 1.0 | ? | 1.0 | 1.0 | ? | ? | 1.0 |
| 9 | ? | ? | 1.0 | ? | 1.0 | ? | ? | 1.0 | 1.0 |

# Voorbeeld simpsons



# Voorbeeld simpsons




| Supp | Conf | Covr | Strg | Lift | Levr | Antecedent | | Consequent |
|---|---|---|---|---|---|---|---|---|
| 0.333 | 1.000 | 0.333 | 1.667 | 1.800 | 0.148 | leeftijd=<30=1.0 | → | gewicht=<=160=1.0 |
| 0.333 | 1.000 | 0.333 | 1.667 | 1.800 | 0.148 | haarlengte=<3=1.0 | → | geslacht=M=1.0 |
| 0.444 | 1.000 | 0.444 | 1.250 | 1.800 | 0.198 | gewicht=>160=1.0 | → | geslacht=M=1.0 |
| 0.444 | 0.800 | 0.556 | 0.800 | 1.800 | 0.198 | geslacht=M=1.0 | → | gewicht=>160=1.0 |
| 0.222 | 1.000 | 0.222 | 2.500 | 1.800 | 0.099 | haarlengte=<3=1.0, gewicht=>160=1.0 | → | geslacht=M=1.0 |
| 0.222 | 1.000 | 0.222 | 2.500 | 1.800 | 0.099 | haarlengte=>5=1.0, gewicht=>160=1.0 | → | geslacht=M=1.0 |
| 0.222 | 1.000 | 0.222 | 2.000 | 2.250 | 0.123 | haarlengte=>5=1.0, geslacht=M=1.0 | → | gewicht=>160=1.0 |
| 0.222 | 1.000 | 0.222 | 2.500 | 1.800 | 0.099 | gewicht=>160=1.0, leeftijd=30-40=1.0 | → | geslacht=M=1.0 |
| 0.222 | 1.000 | 0.222 | 2.000 | 2.250 | 0.123 | leeftijd=30-40=1.0, geslacht=M=1.0 | → | gewicht=>160=1.0 |
| 0.222 | 1.000 | 0.222 | 2.500 | 1.800 | 0.099 | gewicht=>160=1.0, leeftijd=>40=1.0 | → | geslacht=M=1.0 |
| 0.222 | 1.000 | 0.222 | 2.000 | 2.250 | 0.123 | leeftijd=>40=1.0, geslacht=M=1.0 | → | gewicht=>160=1.0 |
| 0.333 | 0.750 | 0.444 | 1.250 | 1.350 | 0.086 | geslacht=V=1.0 | → | haarlengte=>5=1.0 |
| 0.444 | 0.800 | 0.556 | 0.800 | 1.800 | 0.198 | gewicht=<=160=1.0 | → | geslacht=V=1.0 |
| 0.444 | 1.000 | 0.444 | 1.250 | 1.800 | 0.198 | geslacht=V=1.0 | → | gewicht=<=160=1.0 |
| 0.333 | 1.000 | 0.333 | 1.333 | 2.250 | 0.185 | haarlengte=>5=1.0, gewicht=<=160=1.0 | → | geslacht=V=1.0 |
| 0.333 | 0.750 | 0.444 | 1.250 | 1.350 | 0.086 | gewicht=<=160=1.0, geslacht=V=1.0 | → | haarlengte=>5=1.0 |
| 0.333 | 1.000 | 0.333 | 1.667 | 1.800 | 0.148 | haarlengte=>5=1.0, geslacht=V=1.0 | → | gewicht=<=160=1.0 |
| 0.333 | 0.750 | 0.444 | 0.750 | 2.250 | 0.185 | geslacht=V=1.0 | → | haarlengte=>5=1.0, gewicht=<=160=1.0 |
| 0.222 | 1.000 | 0.222 | 2.500 | 1.800 | 0.099 | leeftijd=<30=1.0, geslacht=V=1.0 | → | gewicht=<=160=1.0 |

# Oefeningen

# Oefeningen

- zie Canvas
  - Fruitpromotie
  - Inkomensvragenlijst
  - NASA Website Dataverkeer