

# **DATA WAREHOUSE DAN DATA MINING TUGAS**



**Disusun Oleh :**

Nama: Bisma Aulia

NIM : F1E118006

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS JAMBI**

**2020**

## PROSES ALGORITMA ECLAT

Dalam proses algoritma eclat ini, digunakan dataset belanja pelanggan dengan sampel dibawah ini dengan record data 7501 :

Air Mineral	...	Teh Hijau
Susu		
Pasta Gandum	Kegiatan	
Sup	...	Bawang Merah
Sayuran Beku	...	Teh Hijau
Kentang Goreng		
...	...	...
Telur	...	Susu Yoghurt

Maka, langkah langkah dalam proses algoritma ini adalah :

1. Menampilkan data

```
> data_eclat <- read.csv(file.choose(), header = FALSE)
```

2. Memanggil library arules dan rules viz

```
> library(arules)
> library(arulesviz)
```

3. Memanggil perintah dalam package arules yakni data.transaction, yaitu mentransformasikan data transaksi ke dalam bentuk data transaksi

```
> data_eclat <- read.transactions("D:/Jurnal Data Mining/Algoritma ECLAT/optimisasi_retail.csv", sep = ",", rm.duplicates = TRUE)
distribution of transactions with duplicates:
 1
16
```

4. Melihat ringkasan dari dataset diatas adalah :

```

> summary(data_eclat)
transactions as itemMatrix in sparse format with
7501 rows (elements/itemsets/transactions) and
117 columns (items) and a density of 0.03343942

most frequent items:
  air mineral      telur      spageti
    1788         1348        1306
kentang goreng    coklat    (other)
    1282         1229        22394

element (itemset/transaction) length distribution:
sizes
  1    2    3    4    5    6    7    8    9
1755 1358 1044  815  668  495  390  323  259
 10   11   12   13   14   15   16   18   19
 138  102   67   40   22   17    4    1    2
 20
  1

  Min. 1st Qu.  Median    Mean 3rd Qu.
 1.000  2.000   3.000   3.912  5.000
  Max.
20.000

includes extended item information - examples:
      labels
1      acar
2   air mineral
3 aksesoris pakaian
> itemFrequencyPlot(data_eclat, topN = 10)

```

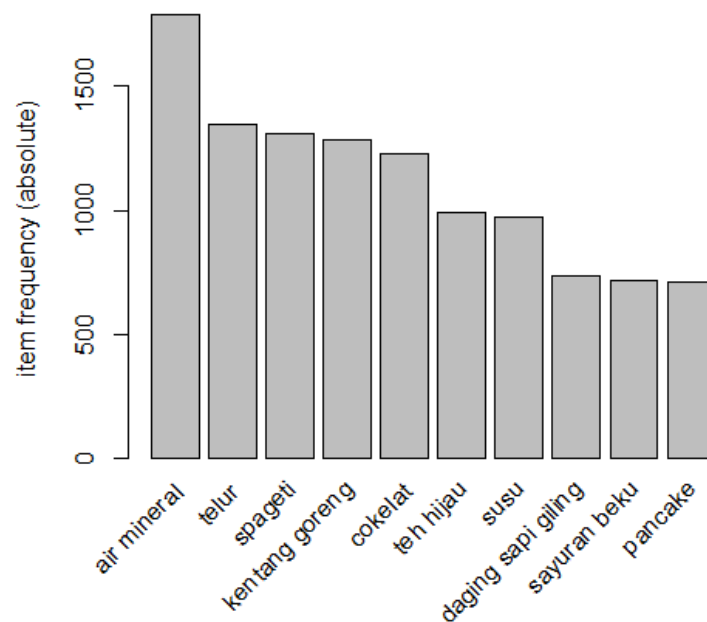
5. Melihat 10 item apa saja yang paling banyak dibeli melalui sebuah histogram adalah :

```

> itemFrequencyPlot(data_eclat, type = "absolute" , topN = 10)

```

Hasilnya adalah



6. Lalu kita akan mentraining datasetnya dan mengaplikasikan algoritma eclat, seperti dibawah ini :

```
> itemFrequencyPlot(data_eclat, topn = 10)
> aturan <- eclat(data = data_eclat, parameter = list(support=0.004, minlen = 2))
Eclat

parameter specification:
tidLists support minlen maxlen
FALSE    0.004    2      10
      target ext
frequent itemsets TRUE

algorithmic control:
sparse sort verbose
  7    -2    TRUE

Absolute minimum support count: 30

create itemset ...
set transactions ... [117 item(s), 7501 transaction(s)] done [0.00s].
sorting and recoding items ... [112 item(s)] done [0.00s].
creating sparse bit matrix ... [112 row(s), 7501 column(s)] done [0.00s].
writing ... [848 set(s)] done [0.01s].
Creating S4 object ... done [0.01s].
```

Dari kodingan diatas kita me-set minimal support nya adalah 0.004 dan minlen / minimal pasangan dalam sebuah transaksi adalah 2 buah / 1 pasangan.

7. Kemudian, kita akan melihat 20 hubungan eclat yang dihasilkan yang diurutkan berdasarkan nilai- supportnya.

```

Creating 34 object ... done [0.013].
> inspect(sort(aturan, by = "support")[1:20])
  items support transIdenticalToItemsets count
[1] {air mineral, spaghetti} 0.05972537 448 448
[2] {air mineral, coklat} 0.05265965 395 395
[3] {air mineral, telur} 0.05092654 382 382
[4] {air mineral, susu} 0.04799360 360 360
[5] {air mineral, daging sapi giling} 0.04092788 307 307
[6] {daging sapi giling, spaghetti} 0.03919477 294 294
[7] {coklat, spaghetti} 0.03919477 294 294
[8] {spageti, telur} 0.03652846 274 274
[9] {kentang goreng, telur} 0.03639515 273 273
[10] {air mineral, sayuran beku} 0.03572857 268 268
[11] {spageti, susu} 0.03546194 266 266
[12] {coklat, kentang goreng} 0.03439541 258 258
[13] {air mineral, pancake} 0.03372884 253 253
[14] {air mineral, kentang goreng} 0.03372884 253 253
[15] {coklat, telur} 0.03319557 249 249
[16] {coklat, susu} 0.03212905 241 241
[17] {air mineral, teh hijau} 0.03106252 233 233
[18] {susu, telur} 0.03079589 231 231
[18] {susu, telur} 0.03079589 231 231
[19] {burger, telur} 0.02879616 216 216
[20] {kentang goreng, teh hijau} 0.02852953 214 214
>

```

Tetapi, apabila tidak dibatasi sebanyak 20 Transaksi, maka, akan menghasilkan 250 hubungan eclat

```

> inspect(sort(aturan, by = "support"))

```

	items	support	transIdenticalToItemsets	count
[1]	{air mineral,spageti}	0.059725370	448	448
[2]	{air mineral,cokelat}	0.052659645	395	395
[3]	{air mineral,telur}	0.050926543	382	382
[4]	{air mineral,susu}	0.047993601	360	360
[5]	{air mineral,daging sapi giling}	0.040927876	307	307
[6]	{daging sapi giling,spageti}	0.039194774	294	294
[7]	{cokelat,spageti}	0.039194774	294	294
[8]	{spageti,telur}	0.036528463	274	274
[9]	{kentang goreng,telur}	0.036395147	273	273
[10]	{air mineral,sayuran beku}	0.035728570	268	268
[11]	{spageti,susu}	0.035461938	266	266
[12]	{cokelat,kentang goreng}	0.034395414	258	258
[13]	{air mineral,pancake}	0.033728836	253	253
[14]	{air mineral,kentang goreng}	0.033728836	253	253
[15]	{cokelat,telur}	0.033195574	249	249
[16]	{cokelat,susu}	0.032129049	241	241
[17]	{air mineral,teh hijau}	0.031062525	233	233
[18]	{susu,telur}	0.030795894	231	231
[19]	{burger,telur}	0.028796161	216	216
[20]	{kentang goreng,teh hijau}	0.028529529	214	214
[21]	{sayuran beku,spageti}	0.027862952	209	209
[22]	{air mineral,minyak zaitun}	0.027596320	207	207
[23]	{kentang goreng,spageti}	0.027596320	207	207
[24]	{air mineral,kue}	0.027463005	206	206
[25]	{spageti,teh hijau}	0.026529796	199	199
[26]	{teh hijau,telur}	0.025463272	191	191
[27]	{pancake,spageti}	0.025196640	189	189
[28]	{air mineral,tomat}	0.024396747	183	183
[29]	{air mineral,burger}	0.024396747	183	183
[30]	{air mineral,susu yogurt}	0.023996800	180	180
[31]	{kentang goreng,susu}	0.023730169	178	178
[32]	{air mineral,udang}	0.023596854	177	177
[33]	{sayuran beku,susu}	0.023596854	177	177
[34]	{cokelat,teh hijau}	0.023463538	176	176

```

> |

```

	items	support	transIdenticalToItemsets	count
[215]	{cokelat,daging sapi giling,spageti}	0.009198773	69	69
[216]	{cokelat,susu,telur}	0.009198773	69	69
[217]	{air mineral,snack batangan}	0.009065458	68	68
[218]	{roti,spageti}	0.009065458	68	68
[219]	{cokelat,rempeh-rempeh}	0.009065458	68	68
[220]	{keju parut,telur}	0.009065458	68	68
[221]	{sup,telur}	0.009065458	68	68
[222]	{teh hijau,tepung beras}	0.009065458	68	68
[223]	{sayuran beku,tepung beras}	0.009065458	68	68
[224]	{ayam,pancake}	0.009065458	68	68
[225]	{minuman rasa,pancake}	0.009065458	68	68
[226]	{burger,kentang goreng,telur}	0.009065458	68	68
[227]	{air mineral,sayuran beku,telur}	0.009065458	68	68
[228]	{daging sapi giling,salmon}	0.008932142	67	67
[229]	{kentang goreng,madu}	0.008932142	67	67
[230]	{minyak zaitun,sup}	0.008932142	67	67
[231]	{burger,tepung beras}	0.008932142	67	67
[232]	{daging tanpa tulang,teh hijau}	0.008932142	67	67
[233]	{ayam,sayuran beku}	0.008932142	67	67
[234]	{daging sapi giling,spageti,telur}	0.008932142	67	67
[235]	{spageti,susu,telur}	0.008932142	67	67
[236]	{air mineral,tuna segar}	0.008798827	66	66
[237]	{pasta gandum,spageti}	0.008798827	66	66
[238]	{bebek,sayuran beku}	0.008798827	66	66
[239]	{burger,udang}	0.008798827	66	66
[240]	{hotdog,spageti}	0.008665511	65	65
[241]	{roti,telur}	0.008665511	65	65
[242]	{daging sapi giling,sayuran beku,spageti}	0.008665511	65	65
[243]	{air mineral,sup,susu}	0.008532196	64	64
[244]	{air mineral,minyak zaitun,susu}	0.008532196	64	64
[245]	{air mineral,spageti,udang}	0.008532196	64	64
[246]	{air mineral,cokelat,kentang goreng}	0.008532196	64	64
[247]	{daging sapi giling,minyak goreng}	0.008398880	63	63
[248]	{kue,tomat}	0.008398880	63	63
[249]	{cokelat,kentang goreng,telur}	0.008398880	63	63
[250]	{air mineral,minyak ikan}	0.008265565	62	62

```

[ reached 'max' / getoption("max.print") -- omitted 598 rows ]
> |

```

8. Untuk memudahkan melihat hubungan dari item tersebut, maka digunakan package `aruleviz` untuk melihat plot berdasarkan nilai supportnya yakni :

```

> plot(aturan[1:20], method = "graph", interactive = TRUE, shading = "support")

```

Hasilnya adalah :

## Graph for 20 itemsets

size: support (0.004 - 0.006)

