

Association Rules Mining

Tugas 3 | PD - A | 2019

KELOMPOK 1

Firin Handayani
Humaira Nur Pradani

05211640000006
05211640000011



PENDAHULUAN

PENGENALAN DATASET	3
--------------------------	---

PRE-STEP

CONVERT FILE .docx MENJADI .csv	4
---------------------------------------	---

BAB I : EKSPLORASI DATA

SUMMARIZATION DATA.....	6
PENJELASAN TIAP ATRIBUT	6
DISTINCT VALUE	6
DIMENSI	7
HEAD & TAIL.....	7
SUMMARY	8
DESCRIBE.....	9
DISTRIBUSI KELAS.....	12
VISUALISASI DATA	13
HEATMAP ATRIBUT NUMERIK	13
MARITAL STATUS	14
AGE.....	16
WORKCLASS	17
OCCUPATION	18
GENDER.....	19
CAPITAL GAIN & LOSS	19
INCOME.....	20

BAB II : PRA-PROSES DATA

DATA DUPLIKAT	21
MISSING VALUE.....	21
Menghilangkan Tanda (?)	22
Membuat 2 Kategori Kelas.....	22
Mengkategorikan Atribut.....	23
Kategori Umur	23
Kategori Jam Kerja	23
Kategori Capital Gain	24
Kategori Capital Loss	25
Kategori Capital Loss	25
Drop Fitur	26



Mengganti Jenis Data Atribut	26
BAB III : IMPLEMENTASI ANALISIS ASOSIASI	
FREQUENT ITEMSET	27
APRIORI	27
Skenario 1.....	27
Skenario 2.....	28
ECLAT	28
Skenario 1.....	29
Skenario 2.....	29
RULES	31
APRIORI	31
Skenario 1.....	31
Skenario 2.....	33
Skenario 3.....	35
Skenario 4.....	36
FP - GROWTH	39
Skenario 1.....	39
Skenario 2.....	41
Skenario 3.....	42
Skenario 4.....	44
BAB IV : HASIL DAN PEMBAHASAN	
Jumlah Rules	46
Support & Confidence.....	47
Interest Factor (Lift)	48
BAB VI : KESIMPULAN	
KESIMPULAN	49



PENGENALAN DATASET

Data yang digunakan diambil dari *the census bureau database* di USA dan diperoleh dari *UCI Machine Learning Repository*. Data tersebut dapat digunakan untuk menganalisis asosiasi antar beberapa atribut non-kelas dengan tingkat penghasilan yang diperoleh. Dataset tersebut memiliki beberapa karakteristik sebagai berikut :



Karakteristik Dataset	Multivariate
Karakteristik Atribut	Categorical, Integer
Associated Tasks	Analisis Asosiasi
Area	Sosial
Number of Instances:	48847
Number of Attributes:	15

Adapun rincian karakteristik tiap atribut adalah sebagai berikut:

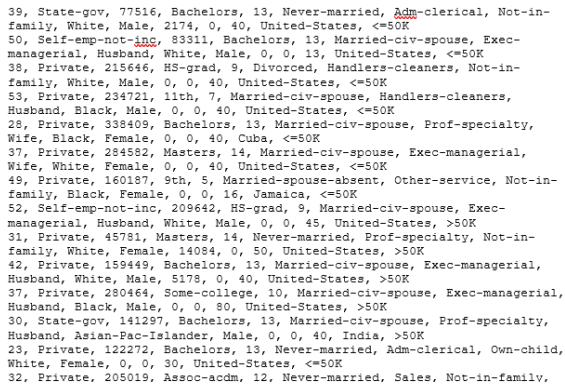
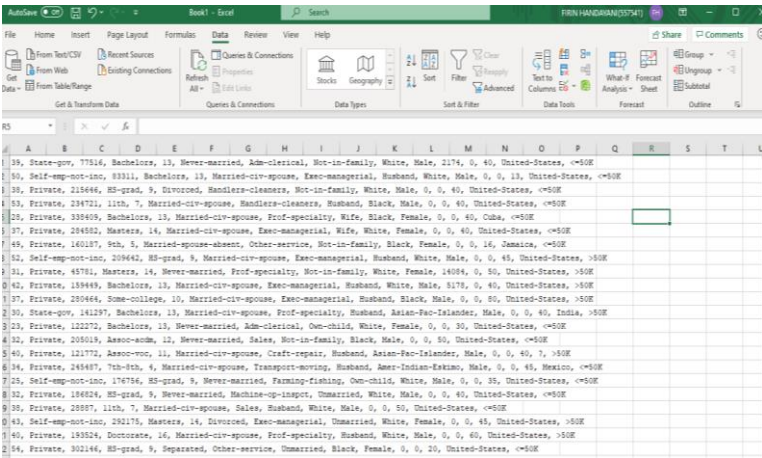
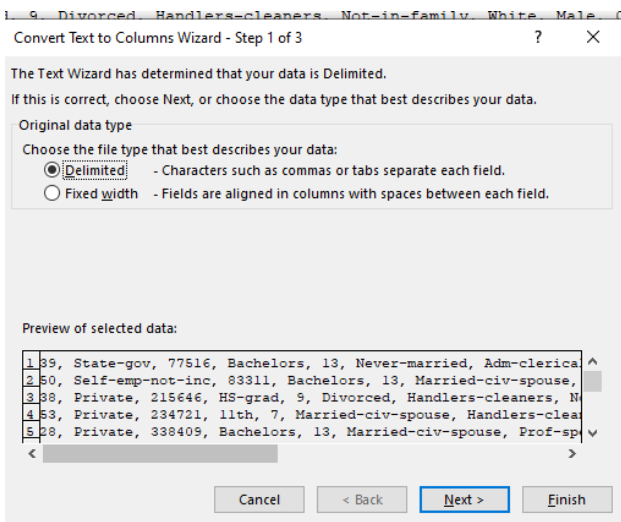
	Karakteristik Dataset	Nilai Atribut
1	Age	continuous
2	workclass	Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
3	fnlwgt	continuous.
4	education	Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
5	education-num	continuous.
6	marital-status	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
7	occupation	Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
8	relationship	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
9	race	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
10	sex	Female, Male.
11	capital-gain	continuous.
12	capital-loss	continuous.
13	hours-per-week	continuous.
14	native-country	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinidad&Tobago, Peru, Hong, Holand-Netherlands.
15	class-label	<=50K, >50K



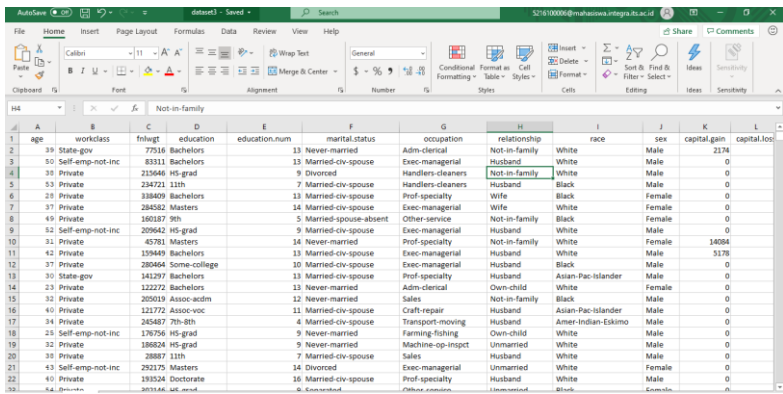
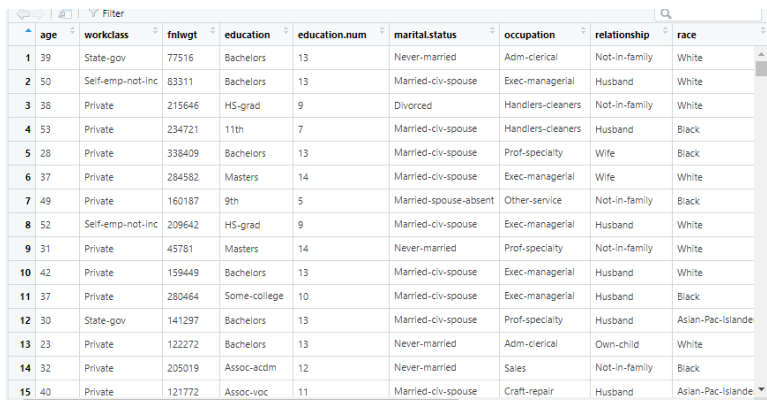
PRE-STEP

CONVERT FILE .docx MENJADI .csv

Pada proses awal perlu dilakukan convert data dari file .docx menjadi file .csv sehingga dapat dilakukan proses eksplorasi data dalam software R.

No.	Gambar	Langkah
1.		<p>Bentuk data awal yang masih dalam format word</p> <p>Copy-Paste data tersebut ke dalam Ms.Excel</p>
2.		<p>Langkah selanjutnya pilih menu "Data" dengan tujuan agar data tersebut terlihat rapi dengan menghilangkan tanda koma</p>
3.		<p>Pada langkah ini pilih "Delimited", hal ini karena data yang kita punya dipisahkan dengan tanda koma yang nantinya akan dijadikan per kolom → klik "Next"</p>



4.	<div> <div> <div>Convert Text to Columns Wizard - Step 2 of 3</div> <div> <div> <div>This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.</div> <div> <div>Delimiters</div> <div> <input type="checkbox"/> Tab <input type="checkbox"/> Semicolon <input type="checkbox"/> Comma <input type="checkbox"/> Space <input checked="" type="checkbox"/> Other: , </div> </div> <div> <input type="checkbox"/> Treat consecutive delimiters as one </div> <div> Text qualifier: " " </div> </div> <div> <div>Data preview</div> <table> <tr><td>39</td><td>State-gov</td><td>77516</td><td>Bachelors</td><td>13</td><td>Never-married</td></tr> <tr><td>50</td><td>Self-emp-not-inc</td><td>83311</td><td>Bachelors</td><td>13</td><td>Married-civ-spouse</td></tr> <tr><td>38</td><td>Private</td><td>215646</td><td>HS-grad</td><td>9</td><td>Divorced</td></tr> <tr><td>53</td><td>Private</td><td>234721</td><td>11th</td><td>7</td><td>Married-civ-spouse</td></tr> <tr><td>28</td><td>Private</td><td>338409</td><td>Bachelors</td><td>13</td><td>Married-civ-spouse</td></tr> </table> <div> <div>Cancel</div> <div>< Back</div> <div>Next ></div> <div>Finish</div> </div> </div> </div> </div> </div>	39	State-gov	77516	Bachelors	13	Never-married	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	38	Private	215646	HS-grad	9	Divorced	53	Private	234721	11th	7	Married-civ-spouse	28	Private	338409	Bachelors	13	Married-civ-spouse	<p>Memisahkan setiap tanda koma dengan kolom sehingga akan terbentuk beberapa kolom yang berisi data seperti gambar di samping.</p> <p>Pilih “Other” dan isi dengan tanda koma (,) kemudian bagian “Text qualifier” isi dengan tanda (") → Klik Next</p>
39	State-gov	77516	Bachelors	13	Never-married																											
50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse																											
38	Private	215646	HS-grad	9	Divorced																											
53	Private	234721	11th	7	Married-civ-spouse																											
28	Private	338409	Bachelors	13	Married-civ-spouse																											
5.	<div>  </div>	<p>Jika data di excel sudah terpisahkan dengan kolom, langkah selanjutnya isi baris pertama dengan judul dari setiap kolom.</p>																														
6.	<div>  </div>	<p>Tampilan di samping merupakan tampilan data saat sudah di import ke dalam software R untuk dilakukan pra-process.</p>																														



SUMMARIZATION DATA

PENJELASAN TIAP ATRIBUT

Dalam melakukan eskplorasi data, hal yang pertama dilakukan yaitu dengan mengetahui karakteristik dari tiap atribut dalam data frame tersebut.

Penggunaan fungsi `str(nama_data_frame)` bertujuan untuk melihat tipe dan struktur dari setiap data frame. Selain itu, fungsi ini akan menampilkan jumlah baris, nama variabel, tipe variabel, dan sebagian baris pertama dari data. Di bawah ini merupakan tampilan penggunaan syntax dan hasil dari struktur data.

Syntax (Karakteristik Tiap Atribut)
<pre>#struktur data str(dataset3)</pre>
Hasil (Karakteristik Tiap Atribut)
<pre>> str(dataset3) Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 48847 obs. of 15 variables: \$ age : num 39 50 38 53 28 37 49 52 31 42 ... \$ workclass : chr "State-gov" "Self-emp-not-inc" "Private" "Private" ... \$ fnlwgt : num 77516 83311 215646 234721 338409 ... \$ education : chr "Bachelors" "Bachelors" "HS-grad" "11th" ... \$ education.num : num 13 13 9 7 13 14 5 9 14 13 ... \$ marital.status : chr "Never-married" "Married-civ-spouse" "Divorced" "Married-civ-spouse" ... \$ occupation : chr "Adm-clerical" "Exec-managerial" "Handlers-cleaners" "Handlers-cleaners" ... \$ relationship : chr "Not-in-family" "Husband" "Not-in-family" "Husband" ... \$ race : chr "white" "white" "white" "Black" ... \$ sex : chr "Male" "Male" "Male" "Male" ... \$ capital.gain : num 2174 0 0 0 0 ... \$ capital.loss : num 0 0 0 0 0 0 0 0 0 ... \$ hours.per.week : num 40 13 40 40 40 40 16 45 50 40 ... \$ cative.country : chr "United-States" "United-States" "United-States" "United-States" ... \$ class.label : chr "<=50K" "<=50K" "<=50K" "<=50K" ... - attr(*, "spec")= .. cols(</pre>

DISTINCT VALUE

Hal selanjutnya setelah mengetahui karakteristik atribut dalam setiap data yaitu melakukan pencarian distinct value dimana hal ini digunakan untuk mencari jumlah nilai yang unik (tidak terdapat duplikat dalam atribut).

Syntax
<pre>#Cari distinct value data movies length(unique(dataset3)) length(unique(dataset3\$age)) length(unique(dataset3\$workclass)) length(unique(dataset3\$fnlwgt)) length(unique(dataset3\$education)) length(unique(dataset3\$education.num)) length(unique(dataset3\$marital.status)) length(unique(dataset3\$occupation)) length(unique(dataset3\$relationship)) length(unique(dataset3\$race)) length(unique(dataset3\$sex)) length(unique(dataset3\$capital.gain)) length(unique(dataset3\$capital.loss)) length(unique(dataset3\$hours.per.week)) length(unique(dataset3\$cative.country)) length(unique(dataset3\$class.label))</pre>
Hasil
<pre>> #Cari distinct value data movies [1] 15 > length(unique(dataset3\$age)) [1] 75 > length(unique(dataset3\$workclass)) [1] 10 > length(unique(dataset3\$fnlwgt)) [1] 28524 > length(unique(dataset3\$cative.country)) [1] 43 > length(unique(dataset3\$class.label)) [1] 5 > length(unique(dataset3\$education)) [1] 17 > length(unique(dataset3\$education.num)) [1] 17 > length(unique(dataset3\$marital.status)) [1] 8 > length(unique(dataset3\$occupation)) [1] 16 > length(unique(dataset3\$relationship)) [1] 7 > length(unique(dataset3\$race)) [1] 6 > length(unique(dataset3\$sex)) [1] 3 > length(unique(dataset3\$capital.gain)) [1] 124 > length(unique(dataset3\$capital.loss)) [1] 100 > length(unique(dataset3\$hours.per.week)) [1] 97</pre>



Hasil dari pencarian distinct value dan penentuan karakteristik setiap atribut terdapat dalam tabel di bawah ini.

Atribut	Distinct Value
age	75
workclass	10
fnlwgt	28524
education	17
education.num	17
marital.status	8
occupation	16
relationship	7
race	6
sex	3
capital.gain	124
capital.loss	100
hours.peer.week	97
cative.country	43
class.label	5

➔ Pencarian distinct value dari setiap atribut dengan menggunakan fungsi `length(unique(nama_data_frame$nama_kolom))`. Penggunaan fungsi ini akan menemukan atribut yang unik dimana tidak ada duplikat atribut dalam data tersebut.

Pada tabel di bawah ini akan menampilkan penggunaan syntax dan hasil pencarian distinct value dan karakteristik dari tiap atribut dengan menggunakan software R.

DIMENSI

Dalam menampilkan jumlah baris dan atribut maka dapat digunakan fungsi `dim(nama_data_frame)`. Atribut dapat diartikan sebagai kolom karena setiap kolom mewakili dari atribut yang ada dalam data frame. Berikut ini merupakan penerapan syntax dan hasilnya dalam menampilkan jumlah baris dan kolom.

Syntax
<pre>#dimensi dim(dataset3)</pre>
Hasil
<pre>> dim(dataset3) [1] 48847 15</pre>

Pada hasil diatas dapat terlihat bahwa jumlah baris dalam data ada 48847 dan jumlah kolom sebanyak 15 kolom.

HEAD & TAIL

Fungsi Head & Tail digunakan untuk menampilkan data. Head bertujuan untuk menampilkan data teratas dari suatu frame dimana biasanya data yang ditampilkan berjumlah 6 ($n=6$). Tail bertujuan untuk menampilkan data terbawah dimana konsepnya sama dengan Head namun hanya berbanding terbalik saja. Data yang ditampilkan berupa 6 baris teratas dan terbawah dari semua atribut yang ada dalam data tersebut.



Penerapan fungsi head & tail → head(nama_data_frame) sedangkan untuk tail yaitu tail(nama_data_frame).

Syntax	
<pre>#head (melihat 6 data teratas) dan tail (melihat 6 data terbawah) head(dataset3) tail(dataset3)</pre>	
Hasil	
<pre>> head(dataset3) # A tibble: 6 x 15 age workclass fnlwtg education education.num marital.status occupation relationship race sex <dbl> <chr> <dbl> <chr> <dbl> <chr> <chr> <chr> <chr> <chr> 1 39 State-gov 77516 Bachelors 13 Never-married Adm-cleri~ Not-in-fami~ white Male 2 50 Self-emp~ 83311 Bachelors 13 Married-civ-s~ Exec-mana~ Husband white Male 3 38 Private 215646 HS-grad 9 Divorced Handlers~ Not-in-fami~ white Male 4 53 Private 234721 11th 7 Married-civ-s~ Handlers~ Husband Black Male 5 28 Private 338409 Bachelors 13 Married-civ-s~ Prof-spec~ wife Black Fema~ 6 37 Private 284582 Masters 14 Married-civ-s~ Exec-mana~ wife white Fema~ # ... with 5 more variables: capital.gain <dbl>, capital.loss <dbl>, hours.per.week <dbl>, # cative.country <chr>, class.label <chr> > tail(dataset3) # A tibble: 6 x 15 age workclass fnlwtg education education.num marital.status occupation relationship race sex <dbl> <chr> <dbl> <chr> <dbl> <chr> <chr> <chr> <chr> <chr> 1 44 Private 83891 Bachelors 13 Divorced Adm-cleri~ Own-child Asia~ Male 2 35 Self-emp~ 182148 Bachelors 13 Married-civ-s~ Exec-mana~ Husband white Male 3 NA NA NA NA NA NA NA NA NA NA 4 NA NA NA NA NA NA NA NA NA NA 5 NA NA NA NA NA NA NA NA NA NA 6 NA NA NA NA NA NA NA NA NA NA # ... with 5 more variables: capital.gain <dbl>, capital.loss <dbl>, hours.per.week <dbl>, # cative.country <chr>, class.label <chr></pre>	

SUMMARY

Dalam menampilkan ringkasan dari suatu data frame dapat menggunakan fungsi summary. Fungsi ini bertujuan untuk menampilkan hasil dari beberapa nilai statistik setiap atribut yang ada di data frame tersebut. Nilai tersebut berupa nilai minimum (Min), nilai quantil pertama (1st Qu.), Nilai tengah (Median), nilai Quantil ketiga (3rd Qu.), dan nilai maksimum.

Penerapan fungsi summary → summary(nama_data_frame).

Syntax	
<pre>#summary -melihat ringkasan data summary(dataset3)</pre>	
Hasil	
<pre>> summary(dataset3) age workclass fnlwtg education education.num Min. :17.00 Length:48847 Min. : 12285 Length:48847 Min. : 1.00 1st Qu.:28.00 Class :character 1st Qu. : 117551 Class :character 1st Qu. : 9.00 Median :37.00 Mode :character Median : 178145 Mode :character Median :10.00 Mean :38.64 Mean : 189664 Mean :10.08 3rd Qu.:48.00 3rd Qu. : 237642 3rd Qu. :12.00 Max. :90.00 Max. :1490400 Max. :16.00 NA's :5 NA's :5 NA's :5 marital.status occupation relationship race sex Length:48847 Length:48847 Length:48847 Length:48847 Length:48847 Class :character Class :character Class :character Class :character Class :character Mode :character Mode :character Mode :character Mode :character Mode :character</pre>	



capital.gain	capital.loss	hours.per.week	cative.country	class.label
Min. : 0	Min. : 0.0	Min. : 1.00	Length:48847	Length:48847
1st Qu.: 0	1st Qu.: 0.0	1st Qu.:40.00	Class :character	Class :character
Median : 0	Median : 0.0	Median :40.00	Mode :character	Mode :character
Mean : 1079	Mean : 87.5	Mean :40.42		
3rd Qu.: 0	3rd Qu.: 0.0	3rd Qu.:45.00		
Max. :99999	Max. :4356.0	Max. :99.00		
NA's :5	NA's :5	NA's :5		

DESCRIBE

Fungsi describe hampir sama dengan summary dimana berguna untuk menampilkan ringkasan dari suatu data frame. Perbedaan dari keduanya yaitu output yang dihasilkan lebih lengkap pada penerapan fungsi describe apalagi untuk data numerik.

Terdapat tampilan informasi missing yang berarti ada tidaknya suatu nilai yang tidak terbaca atau tidak mempunyai nilai (missing value). Selain itu, penggunaan fungsi ini juga dapat menampilkan distinct dimana nilai unik dari setiap atribut dalam data frame. Penerapan sama dimana hanya menuliskan nama dari data frame yang akan ditampilkan.

Syntax

```
#melakukan pendeskripsian data
# : lebih lengkap outputnya drpd summary
describe(dataset3)
```

Hasil

```
> describe(dataset3)
dataset3
```

15	Variables	48847	observations								
----	-----------	-------	--------------	--	--	--	--	--	--	--	--

```
age
```

	n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75
48842		5	74	1	38.64	15.48	19	22	28	37	48
	.90	.95									
	58	63									

```
lowest : 17 18 19 20 21, highest: 86 87 88 89 90
```

```
workclass
```

	n	missing	distinct					
48842		5	9					

```
lowest : ?
highest: Private
```

	Federal-gov	Local-gov	Never-worked	Private	
	Self-emp-inc	Self-emp-not-inc	State-gov	without-pay	
value	?	Federal-gov	Local-gov	Never-worked	Private
Frequency	2799	1432	3136	10	33906
Proportion	0.057	0.029	0.064	0.000	0.694

value	Self-emp-inc	Self-emp-not-inc	State-gov	without-pay
Frequency	1695	3862	1981	21
Proportion	0.035	0.079	0.041	0.000



fnlwgt																
n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75						
48842	5	28523	1	189664	112459	39615	65738	117551	178145	237642						
.90	.95															
328466	379482															
lowest : 12285 13492 13769 13862 14878, highest: 1268339 1366120 1455435 1484705 1490400																

education																
n	missing	distinct														
48842	5	16														
lowest : 10th 11th 12th 1st-4th 5th-6th																
highest: HS-grad Masters Preschool Prof-school Some-college																
10th (1389, 0.028), 11th (1812, 0.037), 12th (657, 0.013), 1st-4th (247, 0.005), 5th-6th (509, 0.010), 7th-8th (955, 0.020), 9th (756, 0.015), Assoc-acdm (1601, 0.033), Assoc-voc (2061, 0.042), Bachelors (8025, 0.164), Doctorate (594, 0.012), HS-grad (15784, 0.323), Masters (2657, 0.054), Preschool (83, 0.002), Prof-school (834, 0.017), Some-college (10878, 0.223)																
=====																
education.num																
n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75						
48842	5	16	0.95	10.08	2.748	5	7	9	10	12						
.90	.95															
13	14															
lowest : 1 2 3 4 5, highest: 12 13 14 15 16																
Value	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Frequency	83	247	509	955	756	1389	1812	657	15784	10878	2061	1601	8025	2657	834	
Proportion	0.002	0.005	0.010	0.020	0.015	0.028	0.037	0.013	0.323	0.223	0.042	0.033	0.164	0.054	0.017	
Value	16															
Frequency	594															
Proportion	0.012															
marital.status																
n	missing	distinct														
48842	5	7														
lowest : Divorced Married-AF-spouse Married-civ-spouse Married-spouse-absent Never-married																
highest: Married-civ-spouse Married-spouse-absent Never-married Separated widowed																
Value	Divorced		Married-AF-spouse		Married-civ-spouse		Married-spouse-absent		Never-married							
Frequency	6633		37		22379		628									
Proportion	0.136		0.001		0.458		0.013									
Value	Never-married		Separated		widowed											
Frequency	16117		1530		1518											
Proportion	0.330		0.031		0.031											

occupation																
n	missing	distinct														
48842	5	15														
lowest : ? Adm-clerical Armed-Forces Craft-repair Exec-managerial																
highest: Prof-specialty Protective-serv Sales Tech-support Transport-moving																
? (2809, 0.058), Adm-clerical (5611, 0.115), Armed-Forces (15, 0.000), Craft-repair (6112, 0.125), Exec-managerial (6086, 0.125), Farming-fishing (1490, 0.031), Handlers-cleaners (2072, 0.042), Machine-op-inspct (3022, 0.062), other-service (4923, 0.101), Priv-house-serv (242, 0.005), Prof-specialty (6172, 0.126), Protective-serv (983, 0.020), Sales (5504, 0.113), Tech-support (1446, 0.030), Transport-moving (2355, 0.048)																

relationship																
n	missing	distinct														
48842	5	6														
lowest : Husband Not-in-family other-relative own-child Unmarried																
highest: Not-in-family Other-relative Own-child Unmarried wife																
Value	Husband	Not-in-family	other-relative	own-child	Unmarried	wife										
Frequency	19716	12583	1506	7581	5125	2331										
Proportion	0.404	0.258	0.031	0.155	0.105	0.048										



```

race
  n missing distinct
48842      5         5

lowest : Amer-Indian-Eskimo Asian-Pac-Islander Black
highest: Amer-Indian-Eskimo Asian-Pac-Islander Black
other
white
white

Value      Amer-Indian-Eskimo Asian-Pac-Islander Black      other      white
Frequency      470      1519      4685      406      41762
Proportion      0.010      0.031      0.096      0.008      0.855

-----

sex
  n missing distinct
48842      5         2

Value      Female      Male
Frequency      16192      32650
Proportion      0.332      0.668

-----

capital.gain
  n missing distinct      Info      Mean      Gmd      .05      .10      .25      .50      .75
48842      5      123      0.228      1079      2086      0      0      0      0      0
.90      .95
0      5013

lowest :      0      114      401      594      914, highest: 25236 27828 34095 41310 99999

-----

capital.loss
  n missing distinct      Info      Mean      Gmd      .05      .10      .25      .50      .75
48842      5      99      0.134      87.5      167.6      0      0      0      0      0
.90      .95
0      0

lowest :      0      155      213      323      419, highest: 3175 3683 3770 3900 4356

-----

hours.per.week
  n missing distinct      Info      Mean      Gmd      .05      .10      .25      .50      .75
48842      5      96      0.897      40.42      12.31      17.05      24.00      40.00      40.00      45.00
.90      .95
55.00      60.00

lowest :      1      2      3      4      5, highest: 95 96 97 98 99

-----

cative.country
  n missing distinct
48842      5      42

lowest : ?
highest: Thailand      Cambodia      Canada      China      Columbia
Trinidad&Tobago      United-States      Vietnam      Yugoslavia

-----

class.label
  n missing distinct
48842      5      4

Value      <=50K      <=50K.      >50K      >50K.
Frequency      24720      12435      7841      3846
Proportion      0.506      0.255      0.161      0.079

```



DISTRIBUSI KELAS

Distribusi kelas bertujuan untuk melihat distribusi maupun presentasi dari setiap kelas suatu data frame. Dalam tabel di bawah ini hanya menampilkan salah satu contoh penerapan distribusi kelas untuk data dengan attribute class.label

Di bawah ini merupakan syntax dan hasil dari distribusi kelas untuk atribut class.label dimana terdapat jumlah dari setiap kelas >50K dan <=50k dan disampingnya ada persentase dari frekuensi kelas tersebut dalam data.

Syntax
<pre>#distribution class y <- dataset3\$class.label cbind(freq=table(y), percentage=prop.table(table(y))*100)</pre>
Hasil
<pre>> cbind(freq=table(y), percentage=prop.table(table(y))*100) freq percentage <=50K 24720 50.61218 <=50K. 12435 25.45965 >50K 7841 16.05381 >50K. 3846 7.87437</pre>



VISUALISASI DATA

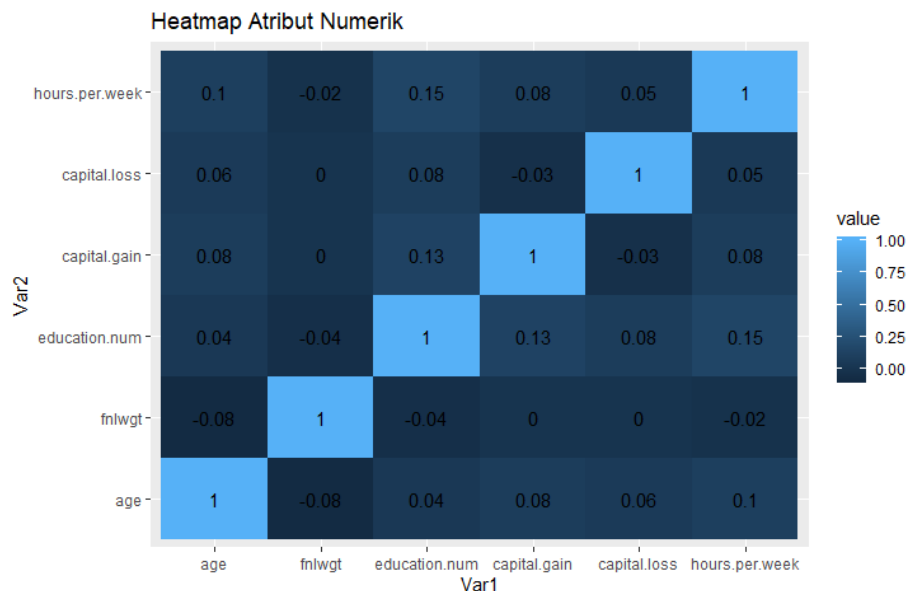
HEATMAP ATRIBUT NUMERIK

Heatmap dibangun dengan menampilkan angka korelasi tiap atribut yang sifatnya numerik. Data numerik pada data tersebut adalah *hours per week*, *capital loss*, *capital gain*, *education.num*, *fnlwgt*, dan *age*.

Syntax

```
#HEATMAP
cor.mat <- round(cor(dataset_clean[,c(1,3,5,11,12,13)]),2)
melted.cor.mat <- melt(cor.mat)
ggplot(melted.cor.mat, aes(x=Var1, y=Var2, fill=value)) + geom_tile() +
  geom_text(aes(x=Var1, y=Var2, label=value)) + ggtitle("Heatmap Atribut Numerik")
```

Hasil



Dari heatmap yang dibangun, terdapat informasi bahwa secara relatif atribut-atribut numerik pada data yang digunakan tidak terlalu berhubungan satu sama lain. Dapat dilihat dari nilai korelasi yang paling tinggi ada pada hubungan *education.num* dan *hours.per.week* yaitu hanya sebesar 0.15. Hal ini berarti kedua atribut tersebut berkorelasi secara positif namun tidak terlalu signifikan. Selain itu, ada beberapa atribut yang sifatnya tidak berkorelasi sama sekali (ditunjukkan dengan nilai korelasi 0). Contoh atribut yang tidak berkorelasi sama sekali adalah *capital.gain* dengan *fnlwgt* dan *capital.loss* dengan *fnlwgt*. Disamping itu ada pula atribut-atribut yang sifatnya berkorelasi negatif, seperti *education.num* dengan *fnlwgt* dan *capital.loss* dengan *capital.gain* serta *fnlwgt* dan *age*. Hal ini berarti dari atribut-atribut tersebut, jika salah satu atribut naik nilainya, maka atribut lainnya akan turun.



MARITAL STATUS

Salah satu atribut yang ada pada data adalah marital status, dimana pada atribut ini dijelaskan tentang status orang-orang terkait pernikahannya.

- **Proporsi pendapatan diatas 50K berdasarkan Marital Status**

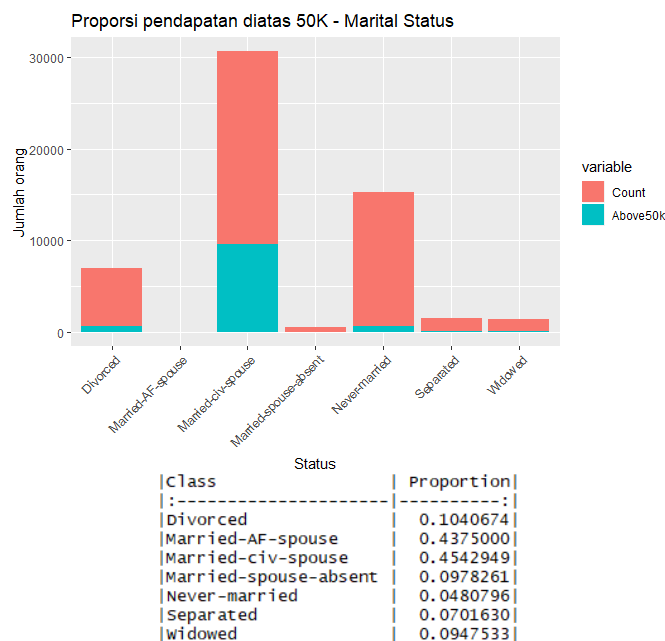
Syntax

```
#proporsi pendapatan diatas 50k - status
library(knitr)
dataset_vis <- dataset
dataset_vis$income<-ifelse(dataset_vis$class.label=='>50K',1,0)
kable(head(dataset_vis))
MaritalLevel<- sqldf("SELECT `marital.status` as status
                      , Count (*) as Count
                      , sum(income) as Above50k
                      FROM dataset_vis
                      GROUP BY status
                      ORDER BY status")

kable(MaritalLevel)

library(reshape2)
Maritalclass<-melt(MaritalLevel,id.vars = 'status')
ggplot(Maritalclass,aes(x=status,y=value,fill=variable))+
  geom_bar(stat = 'identity')+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))+
  ggtitle('Proporsi pendapatan diatas 50K - Marital Status')+
  xlab("Status")+
  ylab("Jumlah orang")
```

Hasil

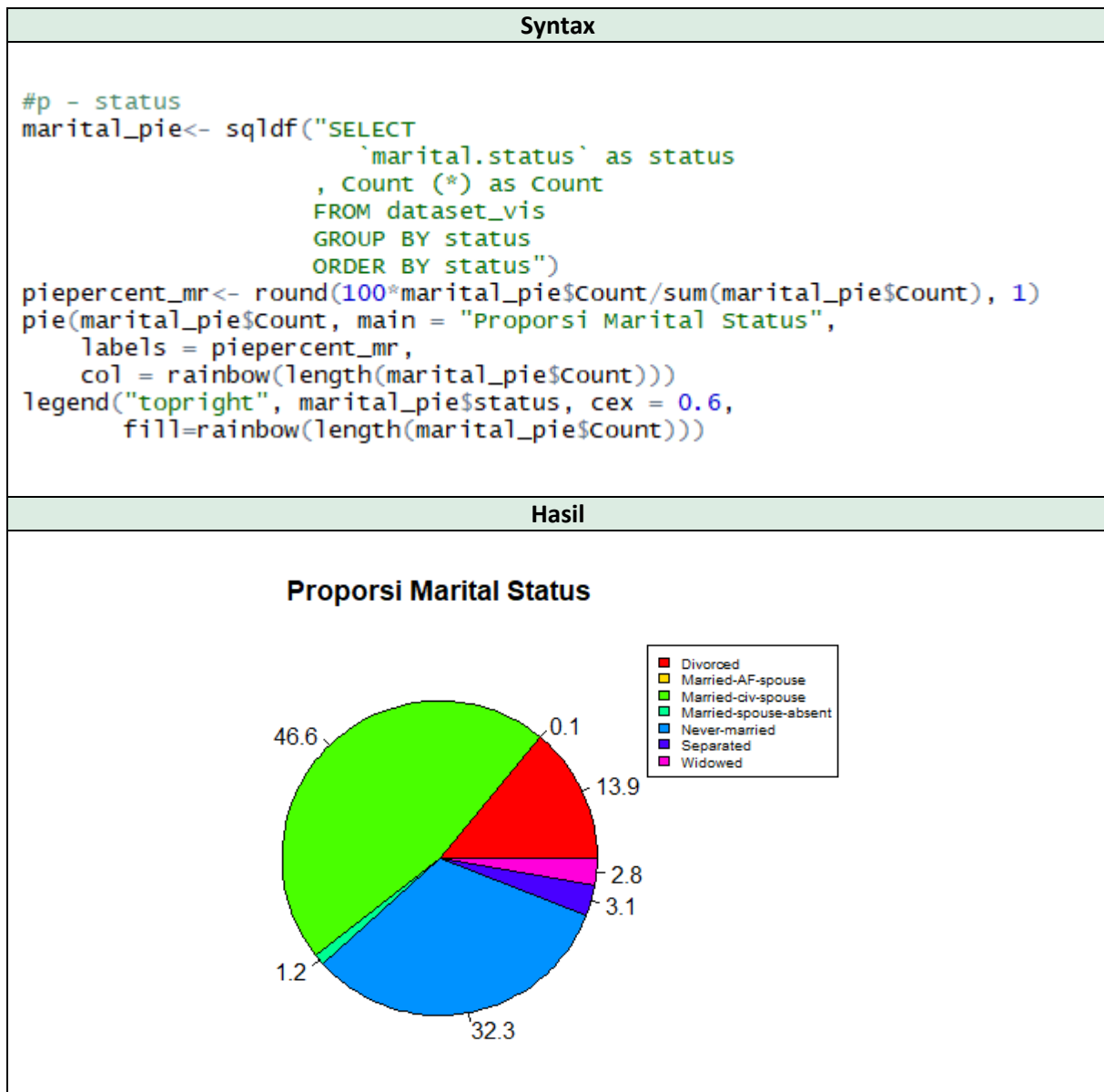


Dari hasil visualisasi tersebut, dapat diketahui bahwa kategori marital status yang paling banyak adalah pada *Married-civ-spouse* yang berarti dari sensus yang telah dilakukan, banyak orang telah menikah dan pasangannya masih hidup. Kemudian, selain itu didapatkan informasi lainnya mengenai proporsi pendapatan per kategorinya. Dari hasil yang didapatkan, proporsi pendapatan diatas 50K yang palig tinggi ada pada individu yang telah menikah dan pasangannya masih hidup (*Married-civ-spouse*).



- **Proporsi masing-masing Marital Status**

Atribut marital status memiliki tujuh kategori yang menjelaskan masing-masing status pernikahan orang yang menjadi responden sensus yang dilakukan. Untuk mengetahui proporsi masing-masing kategori maka syntax R berikut dapat digunakan untuk memvisualisasikan informasi tersebut dalam bentuk Pie Chart. Pie Chart dipilih karena dapat secara baik menampilkan data yang sifatnya kategorikal dan merepresentasikan bagian-dari-keseluruhan suatu data.

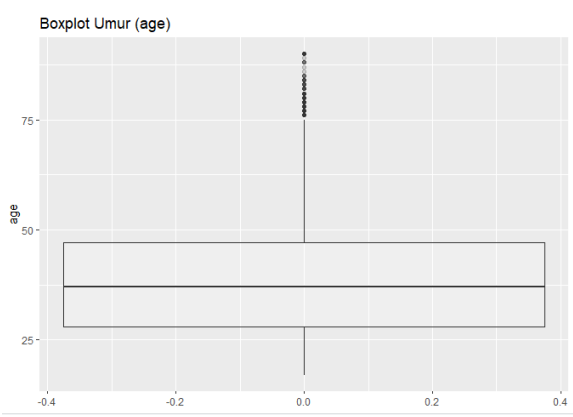
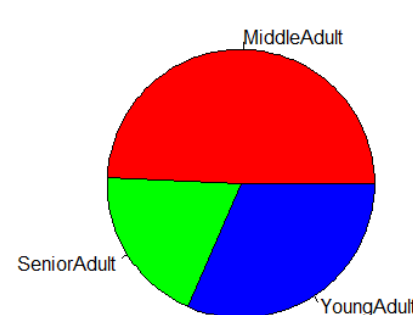


Dari visualisasi tersebut, dapat diketahui bahwa dari sensus yang dilakukan, responden sebanyak 46.6 % berstatus sudah menikah (married-civ-spouse), diikuti sejumlah 32.3% berstatus tidak pernah menikah (never-married), dan 13,9% berstatus cerai (divorced). Porsi terendah sebesar 1,2% adalah status "married-spouse-absent" atau dapat diartikan sudah menikah namun pasangannya telah tiada.



AGE

Atribut lain pada data adalah age dimana atribut ini menyatakan umur dari responden sensus. Distribusi umur dapat dilihat dari boxplot dan pie chart sebagai berikut.

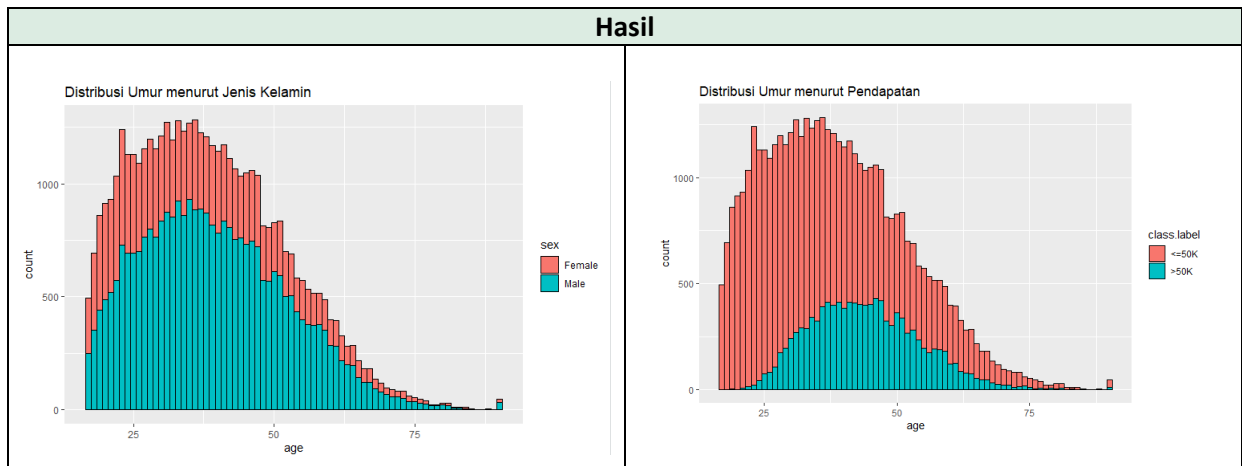
Syntax	
<pre>#boxplot ggplot(dataset_clean, aes(y=age, fill=age)) + geom_boxplot(varwidth = TRUE, alpha=0.2) + theme(legend.position="none") + ggtitle("Boxplot Umur (age)")</pre>	<pre>#p -age age_pie<- sqldf("SELECT age as age , Count (*) as Count FROM dataset_vis GROUP BY age ORDER BY age") piepercent_ag <- round(100*age_pie\$Count/sum(age_pie\$Count), 1) piepercent_ag pie(age_pie\$Count, main = "Proporsi Age", labels = age_pie\$age, col = rainbow(length(age_pie\$Count)))</pre>
Hasil	
	 <pre>> piepercent_ag [1] 49.3 19.2 31.5</pre>

Dari box plot yang dibangun, dapat dilihat bahwa rata-rata umur responden ada pada umur 35. Sedangkan pada pie chart yang terbentuk, data umur digolongkan menjadi 3, yaitu SeniorAdult, MiddleAdult dan YoungAdult. Dari ketiganya, kateogri MiddleAdult memiliki proporsi yang paling tinggi yaitu 49.3%. Responden yang masuk pada kategori ini adalah responden dengan umur antara 31-50 tahun.

Selain itu, persebaran data umur dapat dikaitkan pada beberapa atribut lain seperti besarnya penadpatan dan jenis kelamin. Berikut merupakan syntax yang digunakan untuk membuat histogram yang sesuai untuk menunjukkan informasi tersebut.

Syntax
<pre># histogram of age by income group ggplot(dataset_clean) + aes(x=age, group=class.label, fill=class.label) + geom_histogram(binwidth=1, color='black') +ggtitle("Distribusi Umur menurut Pendapatan") # histogram of age by gender group ggplot(dataset_clean) + aes(x=age, group=sex, fill=sex) + geom_histogram(binwidth=1, color='black') +ggtitle("Distribusi Umur menurut Jenis Kelamin")</pre>

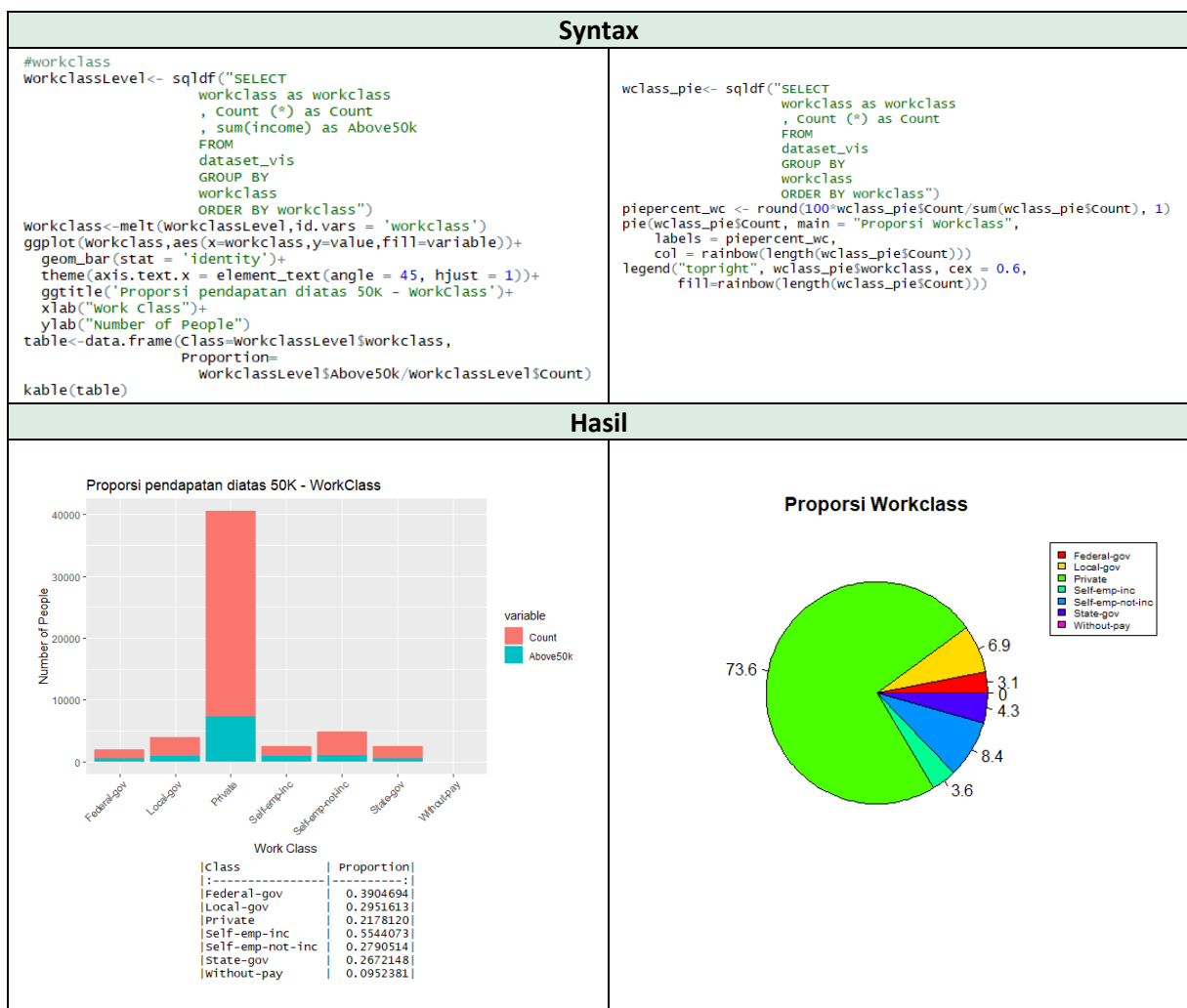




Dari hasil yang didapatkan, dapat dilihat bahwa distribusi umur menurut jenis kelamin bersifat right-skewed atau kemiringan negatif. Sedangkan distribusi umur menurut pendapatan bersifat relative normal pada rata-rata sekitar umur 40.

WORKCLASS

Atribut selanjutnya yaitu workclass menunjukkan pada bidang apa responden bekerja. Proporsi data untuk workclass dapat dilihat pada bar plot dan pie chart berikut.



Dari visualisasi yang dibangun, dapat diketahui bahwa workclass memiliki beberapa kategori seperti Private, Local-gov, Federal-gov, dll. Dari pie chart yang terbentuk dapat diketahui bahwa dari seluruh kategori yang ada, proporsi pekerjaan “private” adalah yang paling besar, yakni 73.6%. Sedangkan, proporsi paling kecil yaitu pada state-government. Sedangkan, barplot yang terbentuk menunjukkan informasi mengenai proporsi orang yang memiliki pendapatan lebih dari 50K berdasarkan workclass. Dapat dilihat bahwa pendapatan diatas 50K tertinggi adalah pada kategori self-emp-inc yaitu sebesar 55.4%. Hal ini berarti, pada pekerjaan berkategori self-emp-inc, lebih dari 50% orangnya memiliki pendapatan diatas 50K.

OCCUPATION

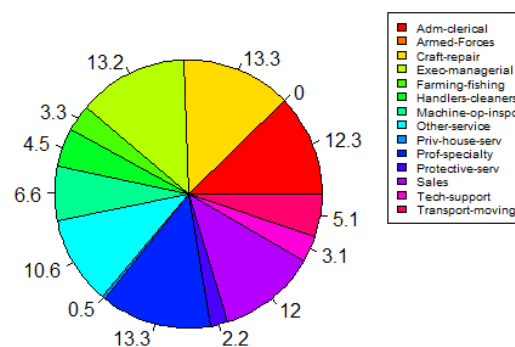
Atribut lainnya yaitu Occupation, dimana atribut ini merepresentasikan pekerjaan yang dilakukan oleh masing-masing responden sensus.

Syntax

```
occ_pie<-sqldf("SELECT
                occupation as occupation
                , Count (*) as Count
                FROM
                dataset_vis
                GROUP BY
                occupation
                ORDER BY occupation")
piepercent_oc <- round(100*occ_pie$Count/sum(occ_pie$Count), 1)
pie(occ_pie$Count, main = "Proporsi Occupation",
    labels = piepercent_oc,
    col = rainbow(length(occ_pie$Count)))
legend("topright", occ_pie$occupation, cex = 0.6,
    fill=rainbow(length(occ_pie$Count)))
```

Hasil

Proporsi Occupation

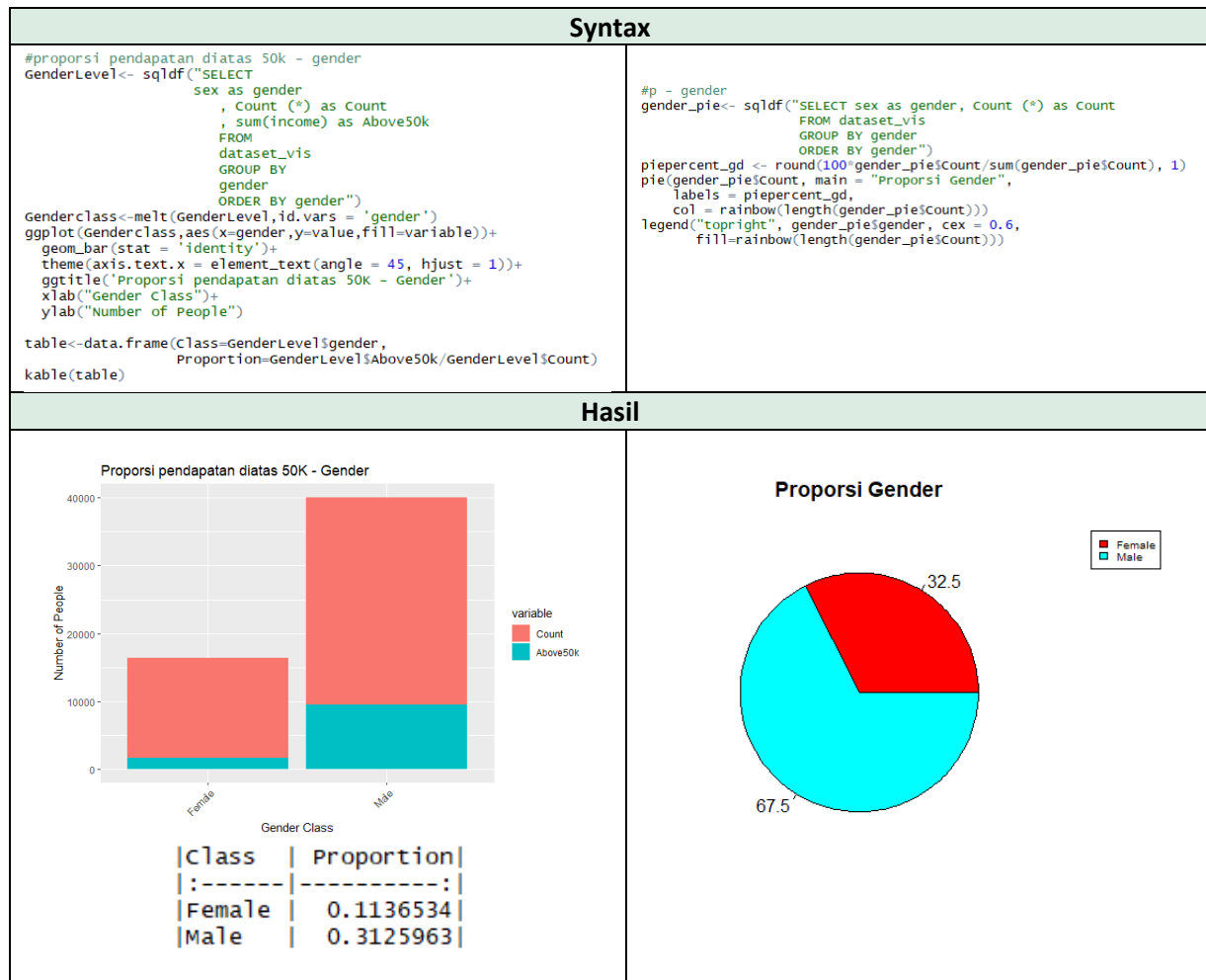


Dari visualisasi diatas, dapat diketahui bahwa ada sekitar 14 kategori pekerjaan, yaitu seperti Adm-clerical, craft-repair, sales, tech-support, dll. Kategori pekerjaan dengan proporsi paling tinggi sebesar 13.3% yaitu craft-repair dan prof-specialty. Hal ini menunjukkan bahwa dari keseluruhan orang yang dissensus, banyak orang memiliki pekerjaan di bidang kerajinan serta perbaikan dan spesialis profesioanl akan pekerjaan tertentu.



GENDER

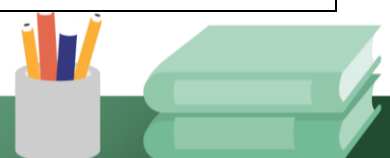
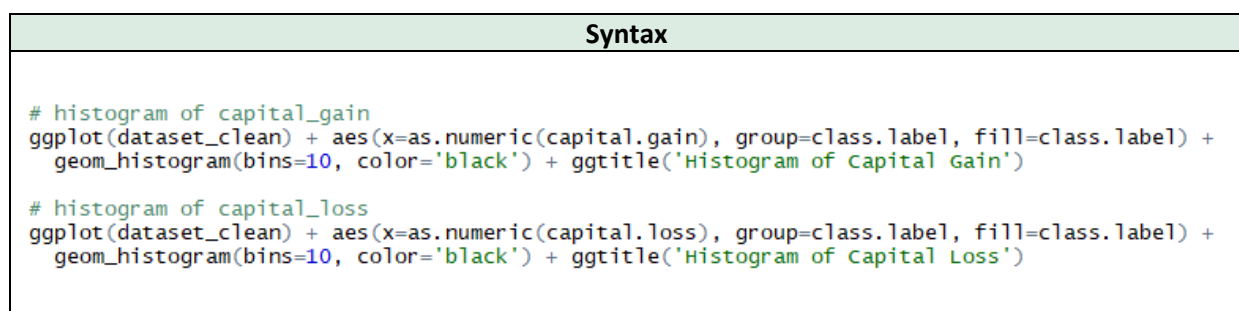
Atribut lainnya yaitu gender atau jenis kelamin. Informasi terkait jenis kelamin tersebut dapat dijabarkan pada visualisasi berbentuk bar plot dan pie chart sebagai berikut.

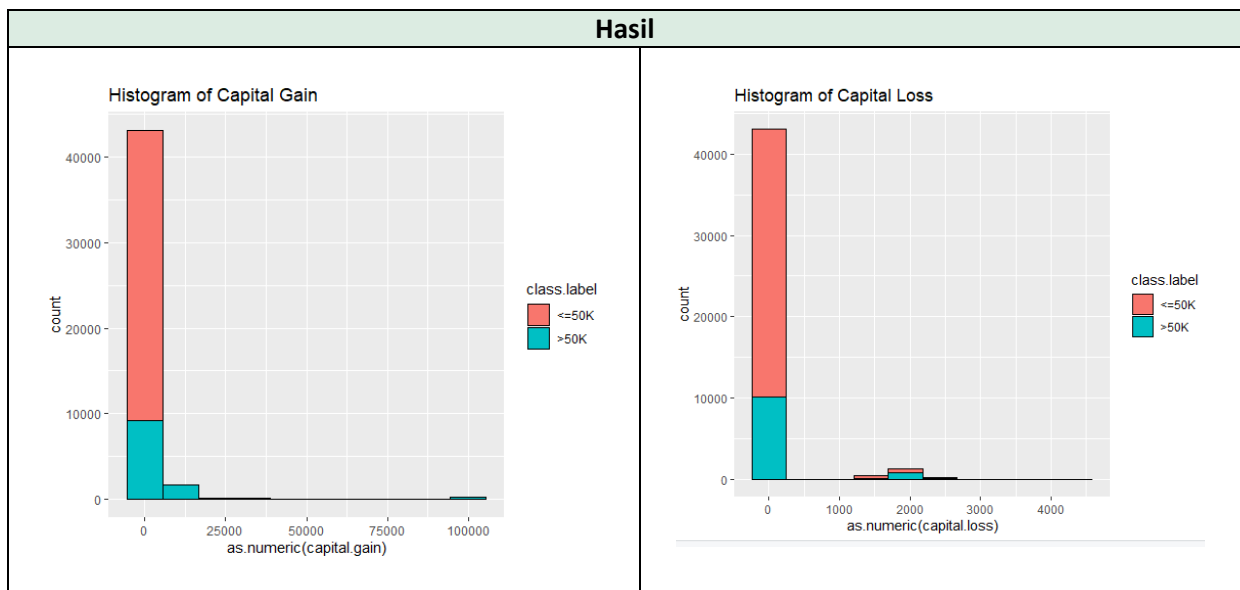


Dari visualisasi yang dibentuk, dapat diketahui bahwa 67.5% responden berjenis kelamin laki-laki dan sisanya berjenis kelamin perempuan. Selain itu, dari barplot yang terbentuk, dapat diketahui pendapatan diatas 50K pada jenis kelamin laki-laki lebih tinggi yakni sebesar 31% dibandingkan perempuan yang hanya 11%.

CAPITAL GAIN & LOSS

Berikut merupakan histogram yang dibangun untuk mengetahui persebaran dari data Capital Gain dan Capital Loss pada data.

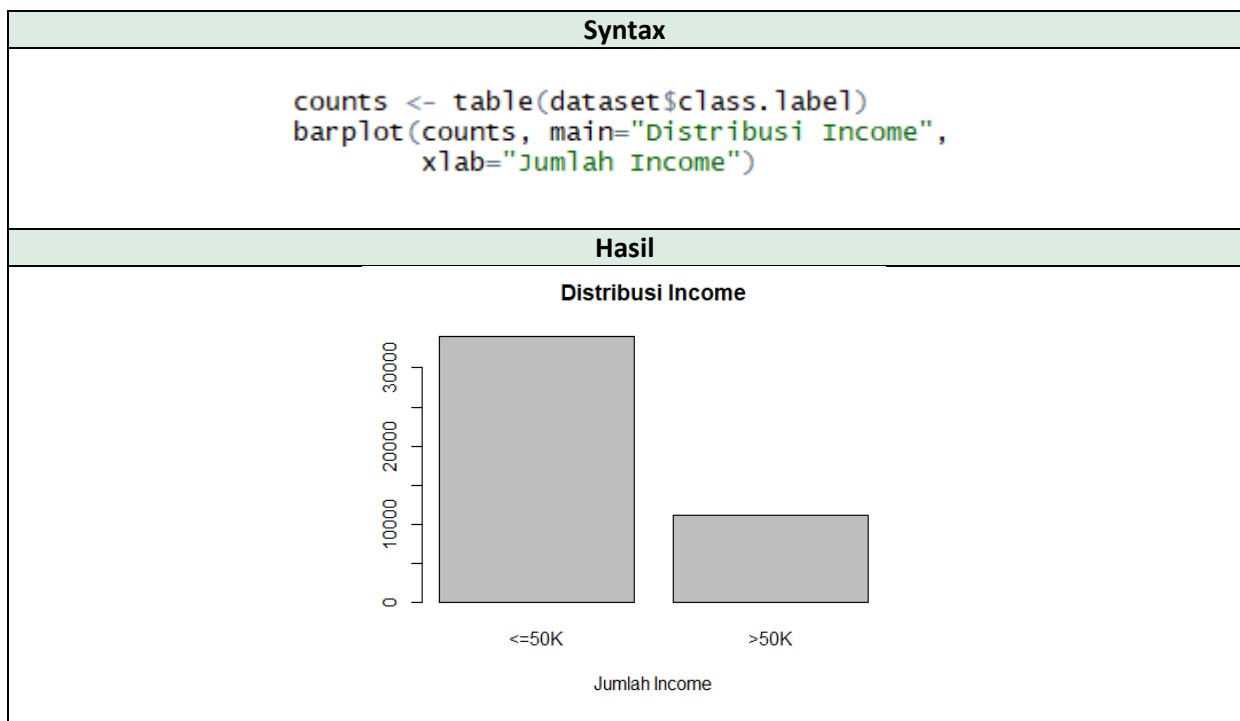




Dari visualisasi yang dibentuk, dapat dilihat bahwa baik capital gain maupun loss memiliki persebaran data yang kurangimbang. Dimana, pada keduanya data yang paling banyak pada range 0-1000, sedangkan pada range lainnya tidak terlalu banyak data. Namun, perbedaannya dapat dilihat bahwa capital gain memiliki batas atas yang lebih tinggi, yaitu pada angka ratusan ribu, sedangkan capital loss memiliki batas atas dibawah 5000.

INCOME

Berikut merupakan histogram yang dibangun untuk mengetahui persebaran dari data pendapatan.



Dari visualisasi yang dibentuk, dapat dilihat bahwa data tersebut memiliki persebaran yang tidak terlalu rata untuk jumlah income (class.label) dimana jumlah data dengan label <=50K jauh lebih banyak dibandingkan dengan data dengan label >50K.



BAB II : PRA-PROSES DATA

DATA DUPLIKAT

Pertama-tama, dalam melakukan pra-proses data, harus ada pengecekan dan penindakanjutan untuk data-data yang sifatnya duplikat atau redundan. Untuk melakukan hal tersebut, diperlukan library dplyr dengan fungsi distinct().

```

Syntax (Pengecekan Data Dupikat)

#data Redundan
datasetr<- dataset3 %>% distinct()

Hasil (Pengecekan Data Dupikat)

Sebelum
dataset3 48847 obs. of 15 variables

Sesudah
datasetr 48814 obs. of 15 variables

```

Dari hasil analisis data duplikat pada dataset tersebut terlihat bahwa dari data awal ada pengurangan jumlah nilai sebanyak 33 baris dimana baris yang hilang merupakan baris yang memiliki nilai yang bersifat redundan atau duplikat.

MISSING VALUE

Hal kedua yang perlu diberi perhatian adalah data-data yang berisi nilai N/A atau NULL. Pengecekan tersebut dapat dilakukan dengan fungsi `is.na()`.

```

Syntax (Missing Value)

#Missing value
#-apakah ada?
any(is.na(datasetr))
#-dimana aja?
sapply(datasetr, function(x) any(is.na(x)))
#-berapa yang N/A?
sapply(datasetr, function(x) sum(is.na(x)))
datasetpd <- datasetr[-c(which(is.na(datasetr))), ]

Hasil (Missing Value)

> #-apakah ada?
> any(is.na(datasetr))
[1] TRUE
> #-dimana aja?
> sapply(datasetr, function(x) any(is.na(x)))
      age      workclass      fnlwgt      education      education.num      marital.status      occupation
relationship      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
      class.label
      TRUE
> #-berapa yang N/A?
> sapply(datasetr, function(x) sum(is.na(x)))
      age      workclass      fnlwgt      education      education.num      marital.status      occupation
relationship      1      1      1      1      1      1      1
      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
      class.label
      1
> datasetpd <- datasetr[-c(which(is.na(datasetr))), ]

```


datasetpd	48813 obs. of 15 variables
datasetr	48814 obs. of 15 variables

Dari hasil diatas yang dapat dilihat bahwa dari datasetr yang awal setelah dilakukan pra-proses data duplikat terlihat ada 1 data yang hilang setelah dilakukan missing value di datasetpd.

Menghilangkan Tanda (?)

Pada data terdapat yang berisi tanda (?) pada beberapa kolom sehingga data yang mengandung data tersebut perlu dihilangkan sehingga data dapat diproses tanpa adanya nilai yang kosong.

Syntax			
<pre>#menghilangkan tanda ? dataset <- datasetpd %>% filter(age != "?", workclass != "?", fnlwt != "?", education != "?", education.num != "?", marital.status != "?", occupation != "?", relationship != "?", race != "?", sex != "?", capital.gain != "?", capital.loss != "?", hours.per.week != "?", cative.country != "?", class.label != "?")</pre>			
Hasil			
Sebelum <table> <tr> <td>datasetpd</td><td>48813 obs. of 15 variables</td></tr> </table>		datasetpd	48813 obs. of 15 variables
datasetpd	48813 obs. of 15 variables		
Sesudah <table> <tr> <td>dataset</td><td>45194 obs. of 15 variables</td></tr> </table>		dataset	45194 obs. of 15 variables
dataset	45194 obs. of 15 variables		

Membuat 2 Kategori Kelas

Pada dataset terdapat kelas yang akan menjadi target masih tergolong 4 kelas dimana kelas tersebut sebenarnya sama yaitu antara >50K dan <=50K. Namun penulisannya saja berbeda yaitu diakhir huruf "K" terdapat tanda titik yang menyebabkan kelas terbagi menjadi 4. Oleh karena itu, diperlukan 2 kelas saja yang dapat mewakili 4 kelas.

Syntax	
<pre>#membuat class.label menjadi 2 kelas (awalnya ada 4) dataset\$class.label[dataset\$class.label == "<=50K."] <- "<=50K" dataset\$class.label[dataset\$class.label == ">50K."] <- ">50K"</pre>	
Hasil	
Sebelum <pre>> length(unique(dataset\$class.label)) [1] 4</pre>	
Sesudah <pre>> length(unique(dataset\$class.label)) [1] 2</pre>	



Mengkategorikan Atribut

Pada langkah ini dilakukan pengkategorian dimana data yang bersifat numerical atau character diubah menjadi data yang bersifat kategori. Tujuan dalam melakukan kategori ini yaitu untuk memudahkan pada saat nanti proses asosiasi.

Kategori Umur

Atribut umur terdiri dari berbagai angka yang akan diubah menjadi 3 kategori saja yaitu :

1. Kategori Young Adult: Orang yang berumur antara 15-30 tahun
2. Kategori Middle Adult: Orang yang berumur antara 31-50 tahun
3. Kategori Senior Adult: Orang yang berumur lebih dari 50 tahun

Kategori umur ini dimulai dari umur 15 tahun karena pada data minimal umur yaitu 17 tahun sehingga tidak perlu menginputkan kategori umur 1-15 tahun.

Syntax

```
#membuat kategori umur
dataset <- (mutate(dataset, age = ifelse(age %in% 15:30, "YoungAdult",
                                         ifelse(age %in% 31:50, "MiddleAdult",
                                                "SeniorAdult"))))
dataset$age <- as.factor(dataset$age)
```

Hasil

Sebelum

age
39
50
38
53
28
37
49
52
31
17

Sesudah

age
MiddleAdult
MiddleAdult
MiddleAdult
MiddleAdult
SeniorAdult
YoungAdult
MiddleAdult
MiddleAdult
SeniorAdult
MiddleAdult
MiddleAdult

Kategori Jam Kerja

Pada atribut jam kerja terdiri dari berbagai angka yang akan diubah menjadi 4 kategori jenis jam kerja berdasarkan rentang angka yang ada di data tersebut. 4 Kategori tersebut yaitu :

1. Kategori Part-Time: Jam kerja yang terdiri dari 1-30 jam per minggu
2. Kategori Full-Time: Jam kerja yang terdiri dari 31-40 jam per minggu
3. Kategori Over-Time: Jam kerja yang terdiri dari 41-56 jam per minggu
4. Kategori Work-Holic: Jam kerja yang lebih dari 56 jam per minggu



Syntax	
<pre>#membuat kategori jam kerja dataset <- (mutate(dataset, hours.per.week = ifelse(hours.per.week %in% 1:30, "Part-Time", ifelse(hours.per.week %in% 31:40, "Full-Time", ifelse(hours.per.week %in% 41:56, "Over-Time", "Work-Holic"))))) dataset\$hours.per.week <- as.factor(dataset\$hours.per.week)</pre>	
Hasil	
Sebelum	Sesudah
hours.per.week	hours.per.week
40	Full-Time
13	Part-Time
40	Full-Time
40	Full-Time
40	Full-Time
40	Full-Time
16	Part-Time
45	Over-Time

Kategori Capital Gain

Atribut capital gain akan diubah menjadi 3 kategori dengan berdasarkan rentang nilai pada data tersebut. 3 Kategori tersebut antara lain:

1. Kategori None: Capital gain yang memiliki nilai 0
2. Kategori Low: Capital gain yang memiliki nilai diantara kurang dari atau sama dengan nilai tengah (Median) dan tidak sama dengan 0
3. Kategori High: Capital gain yang memiliki nilai lebih dari nilai tengah (Median)

Syntax	
<pre>#membuat kategori capital gain dataset[["capital.gain"]] <- ordered(cut(dataset[["capital.gain"]], c(-Inf,0,median(dataset[["capital.gain"]], [dataset[["capital.gain"]]>0]),Inf)), labels=c("None", "Low", "High"))</pre>	
Hasil	
Sebelum	Sesudah
capital.gain	capital.gain
2174	Low
0	None
0	None
0	None
0	None
0	None
0	None
0	None
14084	High



Kategori Capital Loss

Atribut capital loss sama dengan capital gain yaitu data akan diubah menjadi 3 kategori dengan berdasarkan rentang nilai pada data tersebut. 3 Kategori tersebut antara lain:

1. Kategori None: Capital loss yang memiliki nilai 0
2. Kategori Low: Capital loss yang memiliki nilai diantara kurang dari atau sama dengan nilai tengah (Median) dan tidak sama dengan 0
3. Kategori High: Capital ga yang memiliki nilai lebih dari nilai tengah (Median)

Syntax	
<pre>#membuat kategori capital loss dataset[["capital.loss"]] <- ordered(cut(dataset[["capital.loss"]], c(-Inf,0,median(dataset[["capital.loss"]], [dataset[["capital.loss"]]>0]),Inf)), labels=c("None", "Low", "High"))</pre>	
Hasil	
Sebelum	Sesudah
<div>capital.loss</div> <div>0</div> <div>0</div>	<div>capital.loss</div> <div>None</div> <div>None</div>

Kategori Capital Loss

Dalam melakukan asosiasi lebih baik jika semua atribut dibuat kategori. Pada permasalahan ini atribut class.label sudah dibuat menjadi 2 kelas yaitu pendapatan yang $\leq 50K$ dengan $>50K$. Namun untuk menghindari adanya error saat proses asosiasi data tersebut maka dibuat 2 kategori tersebut diubah menjadi tidak angka yaitu:

1. Kategori Low: Pendapatan yang memiliki nilai $\leq 50K$
2. Kategori High: Pendapatan yang memiliki nilai $>50K$

Syntax	
<pre>dataset\$class.label<-ifelse(dataset\$class.label=='>50K',"High","Low")</pre>	
Hasil	
Sebelum	Sesudah
<div>class.label</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$\leq 50K$</div> <div>$>50K$</div>	<div>class.label</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>Low</div> <div>High</div>



Drop Fitur

Pada tahapan ini bertujuan untuk menghilangkan atribut atau fitur yang tidak digunakan dalam proses asosiasi nantinya. Hal ini karena atribut tersebut tidak memiliki hubungan yang erat terhadap apa yang menjadi target dimana dalam case ini yaitu class.label. Atribut yang akan dihilangkan yaitu atribut fnlwgt dan education.num sehingga data akan terdiri dari 13 kolom.

Syntax	
<pre>#drop fitur dataset\$fnlwgt <- NULL dataset\$`education.num` <- NULL</pre>	
Hasil	
Sebelum <pre>> dim(dataset) [1] 45194 15</pre>	Sesudah <pre>> dim(dataset) [1] 45194 13</pre>

Mengganti Jenis Data Atribut

Atribut yang memiliki jenis data berupa character diubah menjadi as.factor semuanya sehingga hal ini dapat mempermudah pada saat proses asosiasi.

Syntax	
<pre>dataset\$workclass <- as.factor(dataset\$workclass) dataset\$education <- as.factor(dataset\$education) dataset\$marital.status <- as.factor(dataset\$marital.status) dataset\$occupation <- as.factor(dataset\$occupation) dataset\$relationship <- as.factor(dataset\$relationship) dataset\$race <- as.factor(dataset\$race) dataset\$sex <- as.factor(dataset\$sex) dataset\$cative.country <- as.factor(dataset\$cative.country) dataset\$class.label <- as.factor(dataset\$class.label)</pre>	
Hasil	
Sebelum <pre>> str(dataset) Classes 'tbl_df', 'tbl' and 'data.frame': 45194 obs. of 13 variables: \$ age : Factor w/ 3 levels "MiddleAdult",...: 1 1 1 2 3 1 1 2 1 1 ... \$ workclass : chr "State-gov" "Self-emp-not-inc" "Private" "Private" ... \$ education : chr "Bachelors" "Bachelors" "HS-grad" "11th" ... \$ marital.status: chr "Never-married" "Married-civ-spouse" "Divorced" "Married-civ-spouse" ... \$ occupation : chr "Adm-clerical" "Exec-managerial" "Handlers-cleaners" "Handlers-cleaners" ... \$ relationship : chr "Not-in-family" "Husband" "Not-in-family" "Husband" ... \$ race : chr "white" "white" "white" "Black" ... \$ sex : chr "Male" "Male" "Male" "Male" ... \$ capital.gain : Ord.factor w/ 3 levels "None"<"Low"<"High": 2 1 1 1 1 1 1 1 3 2 ... \$ capital.loss : Ord.factor w/ 3 levels "None"<"Low"<"High": 1 1 1 1 1 1 1 1 1 1 ... \$ hours.per.week: Factor w/ 4 levels "Full-Time","over-Time",...: 1 3 1 1 1 1 3 2 2 1 ... \$ cative.country: chr "United-States" "United-States" "United-States" "United-States" ... \$ class.label : chr "<=50K" "<=50K" "<=50K" "<=50K" ...</pre>	
Sesudah <pre>> str(dataset) Classes 'tbl_df', 'tbl' and 'data.frame': 45194 obs. of 13 variables: \$ age : Factor w/ 3 levels "MiddleAdult",...: 1 1 1 2 3 1 1 2 1 1 ... \$ workclass : Factor w/ 7 levels "Federal-gov",...: 6 5 3 3 3 3 3 5 3 3 ... \$ education : Factor w/ 16 levels "10th","11th",...: 10 10 12 2 10 13 7 12 13 10 ... \$ marital.status: Factor w/ 7 levels "Divorced","Married-AF-spouse",...: 5 3 1 3 3 3 4 5 3 ... \$ occupation : Factor w/ 14 levels "Adm-clerical",...: 1 4 6 6 10 4 8 4 10 4 ... \$ relationship : Factor w/ 6 levels "Husband","Not-in-family",...: 2 1 2 1 6 6 2 1 2 1 ... \$ race : Factor w/ 5 levels "Amer-Indian-Eskimo",...: 5 5 5 3 3 5 3 5 5 5 ... \$ sex : Factor w/ 2 levels "Female","Male": 2 2 2 2 1 1 1 2 1 2 ... \$ capital.gain : Ord.factor w/ 3 levels "None"<"Low"<"High": 2 1 1 1 1 1 1 1 3 2 ... \$ capital.loss : Ord.factor w/ 3 levels "None"<"Low"<"High": 1 1 1 1 1 1 1 1 1 1 ... \$ hours.per.week: Factor w/ 4 levels "Full-Time","over-Time",...: 1 3 1 1 1 1 3 2 2 1 ... \$ cative.country: Factor w/ 41 levels "Cambodia","Canada",...: 39 39 39 39 5 39 23 39 39 39 ... \$ class.label : Factor w/ 2 levels "<=50K", ">50K": 1 1 1 1 1 1 1 2 2 2 ...</pre>	



FREQUENT ITEMSET

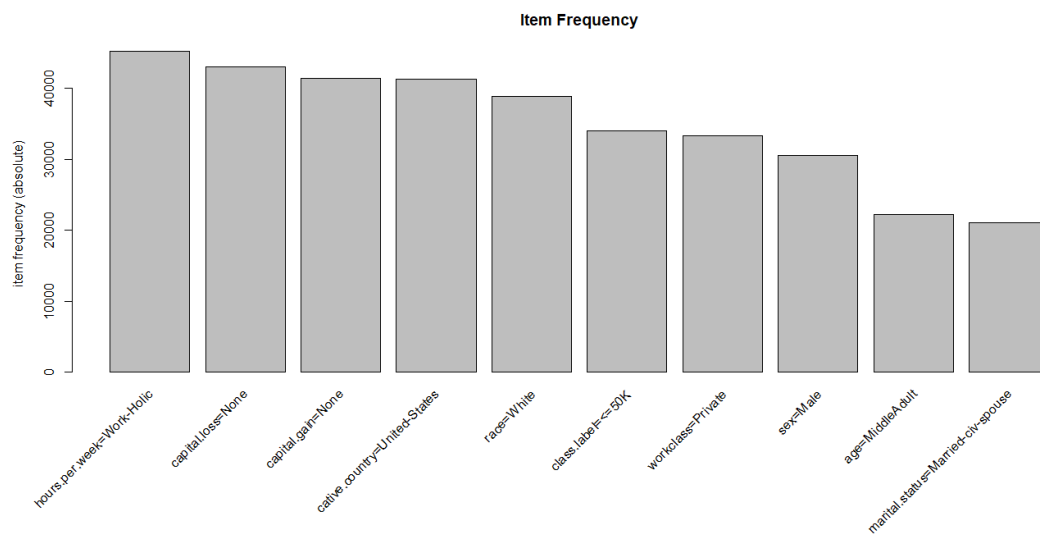
APRIORI

Pertama-tama, kita dapat melihat terlebih dahulu kemunculan-kemunculan tiap item dengan menggunakan function `itemFrequencyPlot()` dari library `arules`.

Syntax

```
dataset.tr<-as(dataset, "transactions")
itemFrequencyPlot(dataset.tr, topN=10, type="absolute", main="Item Frequency")
```

Hasil



Didapatkan bahwa 10 item dengan frekuensi kemunculan tertinggi adalah `hours.per.week=Work-Holic`, dan seterusnya hingga `marital.status=Married-civ-spouse` seperti yang tertera pada gambar.

Dalam membangun frequent itemset dengan menggunakan algoritma apriori kali ini dibagi menjadi 2 skenario yaitu min support 0,2 dan 0,3.

Skenario 1

Pada skenario dengan menggunakan algoritma apriori dimana nilai support = 0.2

Syntax

```
#itemset
itemset <- apriori(datasetapr, parameter=list(support=0.2, minlen=2, target="frequent"))
itemset
sort.itemset<- sort(itemset, by="support")
inspect(sort.itemset[1:10])
```

Hasil

```
Itemsets
> itemset
set of 885 itemsets
10 Itemsets
```



```
> inspect(sort.itemset[1:10])
```

	items	support	count
[1]	{capital.loss=None,cative.country=United-States}	0.8691640	39281
[2]	{capital.gain=None,capital.loss=None}	0.8687879	39264
[3]	{capital.gain=None,cative.country=United-States}	0.8351551	37744
[4]	{race=white,capital.loss=None}	0.8179183	36965
[5]	{race=white,cative.country=United-States}	0.8038014	36327
[6]	{capital.gain=None,capital.loss=None,cative.country=United-States}	0.7911448	35755
[7]	{race=white,capital.gain=None}	0.7855468	35502
[8]	{race=white,capital.loss=None,cative.country=United-States}	0.7633093	34497
[9]	{race=white,capital.gain=None,capital.loss=None}	0.7432403	33590
[10]	{race=white,capital.gain=None,cative.country=United-States}	0.7328185	33119

➔ Dari hasil di atas dapat terlihat bahwa dengan menggunakan algoritma apriori dan support = 0.2 dihasilkan frequent itemsets sebanyak 885. pada gambar di atas hanya ditampilkan 10 itemset dengan mengurutkan nilai support yang tertinggi.

Skenario 2

Pada skenario dengan menggunakan algoritma apriori dimana nilai support = 0.3

Syntax

```
#itemset
itemset <- apriori(datasetapr, parameter=list(support=0.3, minlen=2, target="frequent"))
itemset
sort.itemset<- sort(itemset, by="support")
inspect(sort.itemset[1:10])
```

Hasil

Itemsets

```
> itemset
set of 294 itemsets
10 Itemsets
```

```
> inspect(sort.itemset[1:10])
```

	items	support	count
[1]	{capital.loss=None,cative.country=United-States}	0.8691640	39281
[2]	{capital.gain=None,capital.loss=None}	0.8687879	39264
[3]	{capital.gain=None,cative.country=United-States}	0.8351551	37744
[4]	{race=white,capital.loss=None}	0.8179183	36965
[5]	{race=white,cative.country=United-States}	0.8038014	36327
[6]	{capital.gain=None,capital.loss=None,cative.country=United-States}	0.7911448	35755
[7]	{race=white,capital.gain=None}	0.7855468	35502
[8]	{race=white,capital.loss=None,cative.country=United-States}	0.7633093	34497
[9]	{race=white,capital.gain=None,capital.loss=None}	0.7432403	33590
[10]	{race=white,capital.gain=None,cative.country=United-States}	0.7328185	33119

➔ Dari hasil di atas dapat terlihat bahwa dengan menggunakan algoritma apriori dan support = 0.3 dihasilkan frequent itemsets sebanyak 294. pada gambar di atas hanya ditampilkan 10 itemset dengan mengurutkan nilai support yang tertinggi.

Perbedaan antara kedua skenario 1 dan 2 hanya terletak pada nilai support yang digunakan dimana pada skenario 1 nilai support = 0.2 dapat menghasilkan 885 itemset sedangkan saat menggunakan nilai support = 0.3 dapat menghasilkan 294 itemset. Hal ini dapat terlihat bahwa penambahan nilai support akan semakin mengurangi frequent item set yang dihasilkan.

ECLAT

Algoritma Equivalence Class Transformation (ECLAT) pada permasalahan ini digunakan dalam melakukan pencarian item set yang paling sering muncul. Pada penerapan eclat ini pencarian frequent itemset dengan menggunakan besarnya support tanpa confidence. Selain itu, hasil dari eclat masih



belum menghasilkan rules namun berupa list dari frequent item set berdasarkan data yang sudah dilakukan pra-proses.

Skenario 1

Pada skenario 1 dengan menggunakan nilai support sebesar 0.2

Syntax																																												
<pre>#Frequent Itemset Freq.Itemset <- arules::eclat(data=dataset, parameter=list(supp=0.2)) rules <- sort(Freq.Itemset, decreasing = T, by="supp") rules inspect(rules[1:10])</pre>																																												
Hasil																																												
<pre>Itemsets > rules set of 904 itemsets</pre> <p>10 Itemsets</p> <pre>> inspect(rules[1:10])</pre> <table><thead><tr><th></th><th>items</th><th>support</th><th>count</th></tr></thead><tbody><tr><td>[1]</td><td>{capital.loss=None}</td><td>0.9526486</td><td>43054</td></tr><tr><td>[2]</td><td>{capital.gain=None}</td><td>0.9161393</td><td>41404</td></tr><tr><td>[3]</td><td>{cative.country=United-States}</td><td>0.9131743</td><td>41270</td></tr><tr><td>[4]</td><td>{capital.loss=None,cative.country=United-States}</td><td>0.8691640</td><td>39281</td></tr><tr><td>[5]</td><td>{capital.gain=None,capital.loss=None}</td><td>0.8687879</td><td>39264</td></tr><tr><td>[6]</td><td>{race=white}</td><td>0.8602248</td><td>38877</td></tr><tr><td>[7]</td><td>{capital.gain=None,cative.country=United-States}</td><td>0.8351551</td><td>37744</td></tr><tr><td>[8]</td><td>{race=white,capital.loss=None}</td><td>0.8179183</td><td>36965</td></tr><tr><td>[9]</td><td>{race=white,cative.country=United-States}</td><td>0.8038014</td><td>36327</td></tr><tr><td>[10]</td><td>{capital.gain=None,capital.loss=None,cative.country=United-States}</td><td>0.7911448</td><td>35755</td></tr></tbody></table>		items	support	count	[1]	{capital.loss=None}	0.9526486	43054	[2]	{capital.gain=None}	0.9161393	41404	[3]	{cative.country=United-States}	0.9131743	41270	[4]	{capital.loss=None,cative.country=United-States}	0.8691640	39281	[5]	{capital.gain=None,capital.loss=None}	0.8687879	39264	[6]	{race=white}	0.8602248	38877	[7]	{capital.gain=None,cative.country=United-States}	0.8351551	37744	[8]	{race=white,capital.loss=None}	0.8179183	36965	[9]	{race=white,cative.country=United-States}	0.8038014	36327	[10]	{capital.gain=None,capital.loss=None,cative.country=United-States}	0.7911448	35755
	items	support	count																																									
[1]	{capital.loss=None}	0.9526486	43054																																									
[2]	{capital.gain=None}	0.9161393	41404																																									
[3]	{cative.country=United-States}	0.9131743	41270																																									
[4]	{capital.loss=None,cative.country=United-States}	0.8691640	39281																																									
[5]	{capital.gain=None,capital.loss=None}	0.8687879	39264																																									
[6]	{race=white}	0.8602248	38877																																									
[7]	{capital.gain=None,cative.country=United-States}	0.8351551	37744																																									
[8]	{race=white,capital.loss=None}	0.8179183	36965																																									
[9]	{race=white,cative.country=United-States}	0.8038014	36327																																									
[10]	{capital.gain=None,capital.loss=None,cative.country=United-States}	0.7911448	35755																																									

- ➔ Pada hasil algoritma ECLAT di atas dapat terlihat bahwa terdapat sebanyak 904 itemsets. Setiap baris itemsets masih belum menyertakan target yaitu class.label dimana pendapatan yangb >50K dan <=50K. Dapat disimpulkan bahwa yang dapat mencari frequent itemset saja yaitu algoritma ECLAT dimana pada gambar di atas hanya ditampilkan 10 itemset dengan mengurutkan nilai support yang tertinggi.

Skenario 2

Pada skenario 2 dengan menggunakan nilai support sebesar 0.3

Syntax
<pre>#Frequent Itemset Freq.Itemset <- arules::eclat(data=dataset, parameter=list(supp=0.3)) rules <- sort(Freq.Itemset, decreasing = T, by="supp") rules inspect(rules[1:10])</pre>
Hasil
<pre>Itemsets > rules set of 309 itemsets 10 Itemsets</pre>



```
> inspect(rules[1:10])
```

	items	support	count
[1]	{capital.loss=None}	0.9526486	43054
[2]	{capital.gain=None}	0.9161393	41404
[3]	{cative.country=United-States}	0.9131743	41270
[4]	{capital.loss=None,cative.country=United-States}	0.8691640	39281
[5]	{capital.gain=None,capital.loss=None}	0.8687879	39264
[6]	{race=white}	0.8602248	38877
[7]	{capital.gain=None,cative.country=United-States}	0.8351551	37744
[8]	{race=white,capital.loss=None}	0.8179183	36965
[9]	{race=white,cative.country=United-States}	0.8038014	36327
[10]	{capital.gain=None,capital.loss=None,cative.country=United-States}	0.7911448	35755

- ➔ Pada hasil algoritma ECLAT di atas dapat terlihat bahwa terdapat sebanyak 309 itemsets. Setiap baris itemsets masih belum menyertakan target yaitu class.label dimana pendapatan yangb >50K dan <=50K. Dapat disimpulkan bahwa yang dapat mencari frequent itemset saja yaitu algoritma ECLAT dimana pada gambar di atas hanya ditampilkan 10 itemset dengan mengurutkan nilai support yang tertinggi.

Dari percobaan yang telah dilakukan dapat dilihat bahwa perbedaan antara kedua skenario 1 dan 2 hanya terletak pada nilai support yang digunakan dimana pada skenario 1 nilai support = 0.2 dapat menghasilkan 904 itemset sedangkan saat menggunakan nilai support = 0.3 dapat menghasilkan 309 itemset. Hal ini dapat terlihat bahwa penambahan nilai support akan semakin mengurangi frequent item set yang dihasilkan.



RULES APRIORI

Algoritma apriori merupakan salah satu algoritma klasik data mining. Algoritma apriori digunakan agar komputer dapat mempelajari aturan asosiasi, mencari pola hubungan antar satu atau lebih item dalam suatu dataset. Algoritma apriori banyak digunakan pada data transaksi atau biasa disebut market basket, misalnya sebuah swalayan memiliki market basket, dengan adanya algoritma apriori, pemilik swalayan dapat mengetahui pola pembelian seorang konsumen, jika seorang konsumen membeli item A , B, punya kemungkinan 50% dia akan membeli item C, pola ini sangat signifikan dengan adanya data transaksi selama ini. Penting tidaknya suatu aturan asosiatif dapat diketahui dengan dua parameter, support (nilai penunjang) yaitu persentase kombinasi item tersebut dalam database dan confidence (nilai kepastian) yaitu kuatnya hubungan antar item dalam aturan asosiatif.

Pada permasalahan ini, penerapan apriori terbagi menjadi 4 skenario pembentukan rules dimana membandingkan perbedaan pada nilai support dan confidence dari 4 rules tersebut. Penentuan minimal support dan minimal confidence diharapkan menghasilkan rules yang memiliki support dan confidence yang tinggi sehingga rules yang dihasilkan dapat menjadi rules yang menarik. Berikut merupakan daftar skenario yang digunakan :

Skenario	Min. Support	Min. Confidence
1	0.2	0.6
2	0.3	0.7
3	0.2	0.6
4	0.3	0.7

Dari skenario tersebut, kemudian dijalankan algoritma apriori untuk masing-masing skenario yang telah ditentukan sebelumnya sebagai berikut :

Skenario 1

Pada skenario 1 menggunakan min support = 0,2 dan min confidence = 0.6

Syntax
<pre>library(arules) #Rules1 Rules1_apri <- apriori(dataset, control = list(verbose=F), parameter = list(minlen=2, supp=0.2, conf=0.6), appearance = list(rhs=c("class.label=High", "class.label=Low"), default="lhs")) Rules1_apri inspect(Rules1_apri) #Sort Rules1 sort.rule1 <- sort(Rules1_apri, by="lift") inspect(sort.rule1) #VisualizationRules1 plot(sort.rule1[1:10], method="graph", control=list(nodecol="red", edgecol="blue")) plot(sort.rule1) plot(sort.rule1, method="grouped", control=list(col=2))</pre>
Hasil
<p>Jumlah Rules</p> <pre>> Rules1_apri set of 254 rules</pre>



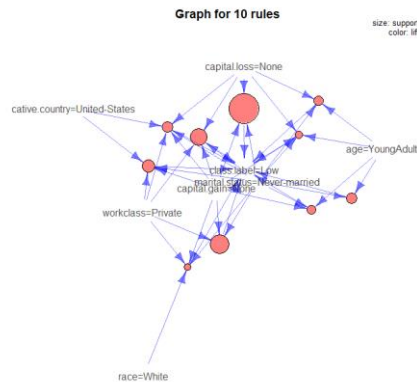
Rules

```
> inspect(sort.rule1[1:5])
```

	lhs	rhs	support	confidence	lift	count
[1]	{age=YoungAdult, marital.status=Never-married, capital.gain=None, capital.loss=None}	=> {class.label=Low}	0.2067974	0.9888901	1.314932	9346
[2]	{age=YoungAdult, marital.status=Never-married, capital.gain=None}	=> {class.label=Low}	0.2117316	0.9875129	1.313100	9569
[3]	{age=YoungAdult, marital.status=Never-married, capital.loss=None}	=> {class.label=Low}	0.2124397	0.9834067	1.307640	9601
[4]	{age=YoungAdult, marital.status=Never-married}	=> {class.label=Low}	0.2173740	0.9822036	1.306041	9824
[5]	{workclass=Private, marital.status=Never-married, capital.gain=None, capital.loss=None}	=> {class.label=Low}	0.2408948	0.9738796	1.294972	10887

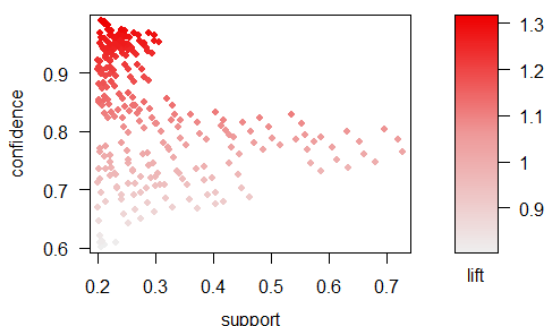
Aturan yang dihasilkan dengan skenario ini adalah berjumlah 254 dengan aturan dengan nilai lift tertinggi yaitu jika capital.gain=None, capital.loss=None, marital.status=Never-married, dan age=YoungAdult maka incomenya termasuk Low ($\leq 50K$). Selain itu, ada 253 aturan asosiasi lain yang dihasilkan.

Graph Rules 1-10



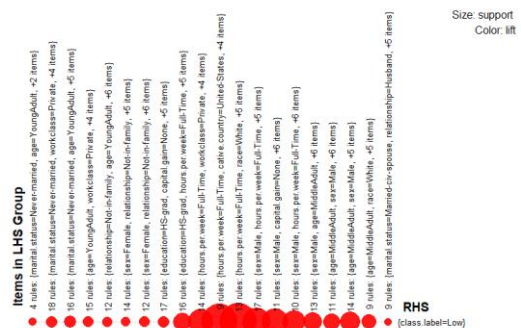
Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

Scatter plot for 254 rules



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift

Grouped Matrix for 254 Rules



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang



direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan low ($\leq 50K$). Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel low.

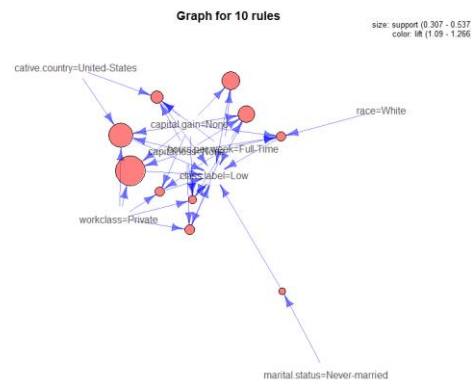
Skenario 2

Pada skenario 2 menggunakan min support = 0.3 dan min confidence = 0.6

Syntax																																										
<pre>#Rules2 Rules2_apri <- apriori(dataset, control = list(verbose=F), parameter = list(minlen=2, supp=0.3, conf=0.6), appearance = list(rhs=c("class.label=High", "class.label=Low"), default="lhs")) Rules2_apri inspect(Rules2_apri) #Sort Rules2 sort.rule2 <- sort(Rules2_apri, by="lift") inspect(sort.rule2[1:5]) #VisualizationRules2 plot(sort.rule2[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue")) plot(sort.rule2) plot(sort.rule2, method="grouped", control=list(col=2))</pre>																																										
Hasil																																										
<p>Jumlah Rules</p> <pre>> Rules2_apri set of 78 rules</pre>																																										
<p>Rules</p> <pre>> inspect(sort.rule2[1:5])</pre> <table> <thead> <tr> <th></th><th>lhs</th><th>rhs</th><th>support</th><th>confidence</th><th>lift</th><th>count</th></tr> </thead> <tbody> <tr> <td>[1]</td><td>{marital.status=Never-married}</td><td>=> {class.label=Low}</td><td>0.3070983</td><td>0.9519204</td><td>1.265773</td><td>13879</td></tr> <tr> <td>[2]</td><td>{workclass=Private, capital.gain=None, capital.loss=None, hours.per.week=Full-Time}</td><td>=> {class.label=Low}</td><td>0.3201531</td><td>0.8549902</td><td>1.136884</td><td>14469</td></tr> <tr> <td>[3]</td><td>{workclass=Private, capital.gain=None, hours.per.week=Full-Time}</td><td>=> {class.label=Low}</td><td>0.3296898</td><td>0.8437624</td><td>1.121955</td><td>14900</td></tr> <tr> <td>[4]</td><td>{capital.gain=None, capital.loss=None, hours.per.week=Full-Time}</td><td>=> {class.label=Low}</td><td>0.4047882</td><td>0.8319236</td><td>1.106213</td><td>18294</td></tr> <tr> <td>[5]</td><td>{workclass=Private, capital.loss=None, hours.per.week=Full-Time}</td><td>=> {class.label=Low}</td><td>0.3332743</td><td>0.8283562</td><td>1.101469</td><td>15062</td></tr> </tbody> </table>		lhs	rhs	support	confidence	lift	count	[1]	{marital.status=Never-married}	=> {class.label=Low}	0.3070983	0.9519204	1.265773	13879	[2]	{workclass=Private, capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3201531	0.8549902	1.136884	14469	[3]	{workclass=Private, capital.gain=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3296898	0.8437624	1.121955	14900	[4]	{capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.4047882	0.8319236	1.106213	18294	[5]	{workclass=Private, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3332743	0.8283562	1.101469	15062
	lhs	rhs	support	confidence	lift	count																																				
[1]	{marital.status=Never-married}	=> {class.label=Low}	0.3070983	0.9519204	1.265773	13879																																				
[2]	{workclass=Private, capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3201531	0.8549902	1.136884	14469																																				
[3]	{workclass=Private, capital.gain=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3296898	0.8437624	1.121955	14900																																				
[4]	{capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.4047882	0.8319236	1.106213	18294																																				
[5]	{workclass=Private, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3332743	0.8283562	1.101469	15062																																				
<p>Aturan yang dihasilkan dengan skenario ini adalah berjumlah 78 dengan aturan dengan nilai lift tertinggi yaitu jika marital.status=Never-married maka incomenya termasuk Low ($\leq 50K$). Selain itu, ada 77 aturan asosiasi lain yang dihasilkan.</p>																																										

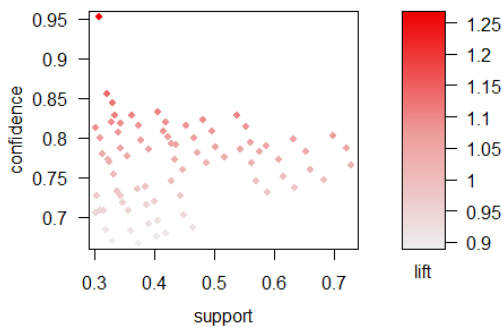


Graph Rules 1:10



Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

Scatter plot for 78 rules



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Grouped Matrix for 78 Rules



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan low ($\leq 50K$). Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel low.



Skenario 3

Pada skenario 3 menggunakan min support = 0,2 dan min confidence = 0.7

Syntax

```
#Rules3
Rules3_apri <- apriori(dataset, control = list(verbose=F),
                      parameter = list(minlen=2, supp=0.2, conf=0.7),
                      appearance = list(rhs=c("class.label=High", "class.label=Low"), default="lhs"))

Rules3_apri
inspect(Rules3_apri)

#Sort Rules3
sort.rule3 <- sort(Rules3_apri, by="lift")
inspect(sort.rule3[1:5])

#VisualizationRules3
plot(sort.rule3[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue"))
plot(sort.rule3)
plot(sort.rule3, method="grouped", control=list(col=2))
```

Hasil

Jumlah Rules

```
> Rules3_apri
set of 222 rules
```

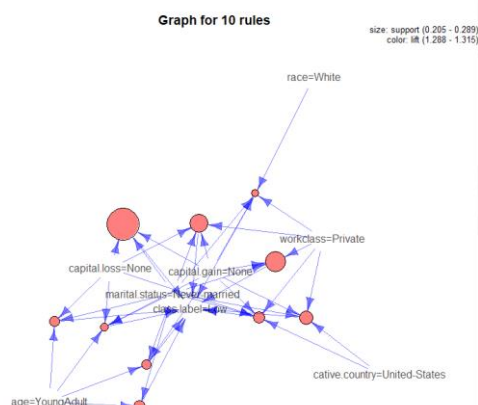
Rules

```
> inspect(sort.rule3[1:5])
```

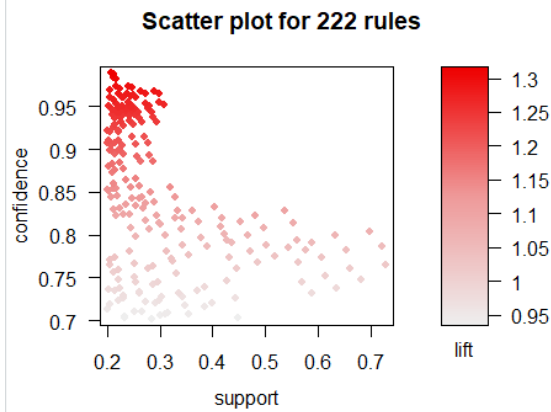
	lhs	rhs	support	confidence	lift	count
[1]	{age=YoungAdult, marital.status=Never-married, capital.gain=None, capital.loss=None}	=> {class.label=Low}	0.2067974	0.9888901	1.314932	9346
[2]	{age=YoungAdult, marital.status=Never-married, capital.gain=None}	=> {class.label=Low}	0.2117316	0.9875129	1.313100	9569
[3]	{age=YoungAdult, marital.status=Never-married, capital.loss=None}	=> {class.label=Low}	0.2124397	0.9834067	1.307640	9601
[4]	{age=YoungAdult, marital.status=Never-married}	=> {class.label=Low}	0.2173740	0.9822036	1.306041	9824
[5]	{workclass=Private, marital.status=Never-married, capital.gain=None, capital.loss=None}	=> {class.label=Low}	0.2408948	0.9738796	1.294972	10887

Aturan yang dihasilkan dengan skenario ini adalah berjumlah 222 dengan aturan dengan nilai lift tertinggi yaitu jika capital.gain=None, capital.loss=None, marital.status=Never-married, dan age=YoungAdult maka incomenya termasuk Low ($\leq 50K$). Selain itu, ada 221 aturan asosiasi lain yang dihasilkan.

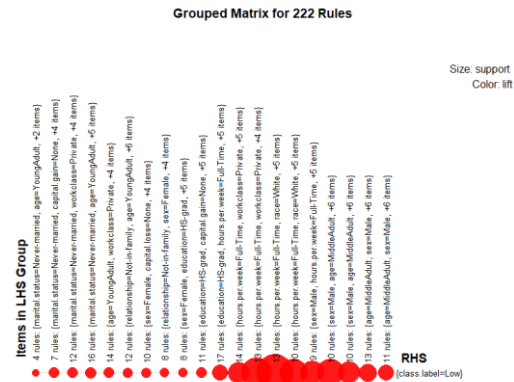
Graph Rules 1-10



Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan low ($\leq 50K$). Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel low.

Skenario 4

Pada skenario 4 menggunakan min support = 0,3 dan min confidence = 0.7

Syntax

```
#Rules4
Rules4_apri <- apriori(dataset, control = list(verbose=F),
                      parameter = list(minlen=2, supp=0.3, conf=0.7),
                      appearance = list(rhs=c("class.label=High", "class.label=Low"), default="lhs"))

Rules4_apri
inspect(Rules4_apri)

#Sort Rules4
sort.rule4 <- sort(Rules4_apri, by="lift")
inspect(sort.rule4[1:5])

#VisualizationRules4
plot(sort.rule4[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue"))
plot(sort.rule4)
plot(sort.rule4, method="grouped", control=list(col=2))
```

Hasil

Jumlah Rules

```
> Rules4_apri
set of 69 rules
```



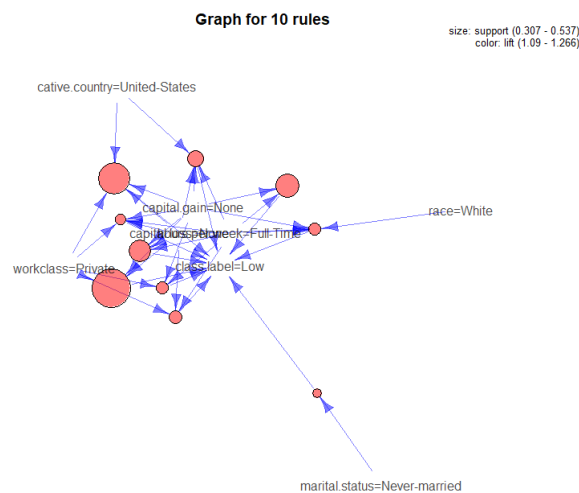
Rules

```
> inspect(sort.rule4[1:5])
```

	lhs	rhs	support	confidence	lift	count
[1]	{marital.status=Never-married}	=> {class.label=Low}	0.3070983	0.9519204	1.265773	13879
[2]	{workclass=Private, capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3201531	0.8549902	1.136884	14469
[3]	{workclass=Private, capital.gain=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3296898	0.8437624	1.121955	14900
[4]	{capital.gain=None, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.4047882	0.8319236	1.106213	18294
[5]	{workclass=Private, capital.loss=None, hours.per.week=Full-Time}	=> {class.label=Low}	0.3332743	0.8283562	1.101469	15062

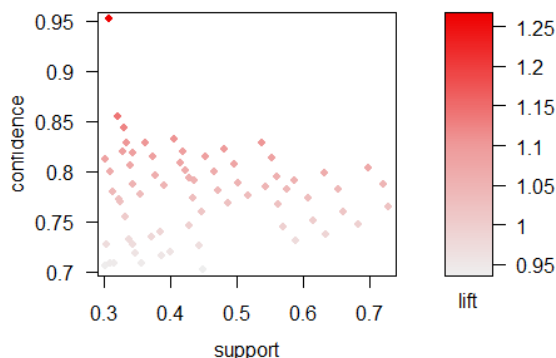
Aturan yang dihasilkan dengan skenario ini adalah berjumlah 69 dengan aturan dengan nilai lift tertinggi yaitu jika marital.status=Never-married maka incomenya termasuk Low ($\leq 50K$). Selain itu, ada 68 aturan asosiasi lain yang dihasilkan.

Graph Rules 1-10



Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

Scatter plot for 69 rules



Grouped Matrix for 69 Rules



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan low ($\leq 50K$). Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel low.



FP - GROWTH

Fp-growth adalah sebuah metode dalam data mining untuk mencari frequent itemset tanpa menggunakan candidate generation. Fp-growth ini merupakan pengembangan dari algoritma apriori. Kekurangan dari algoritma apriori diperbaiki dengan menghilangkan candidate generation, karena Fp-growth menggunakan konsep pembangunan tree dalam pencarian frequent item-set. Hal tersebut membuat algoritma Fp-growth lebih cepat dibandingkan algoritma apriori.

Skenario untuk analisis asosiasi dengan menggunakan algoritma fp-growth disamakan seperti penerapan algoritma apriori sebelumnya agar nantinya dapat dibandingkan hasil dari kedua algoritma yang telah dijalankan.

Skenario 1

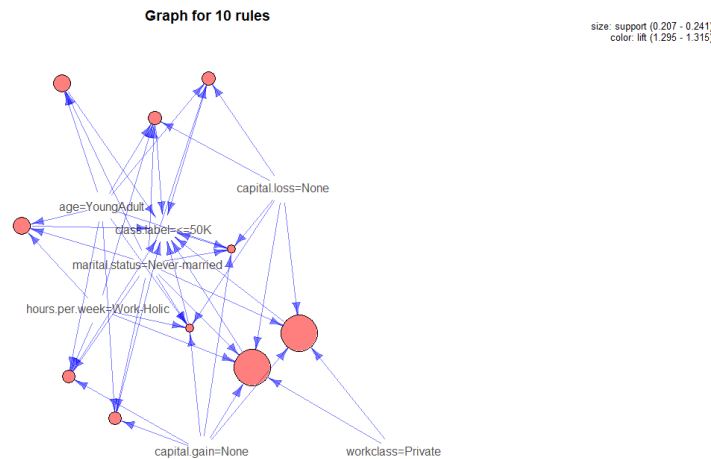
Skenario 1 menggunakan nilai support minimum 0,2 dan confidence 0,6 serta maxlength = 8.

Syntax					
<pre>#rules1===== rules1_fp <- rCBA::fpgrowth(dataset, support = 0.2, confidence = 0.6, maxLength = 8, consequent = "class.label", parallel=FALSE) rules1_fp inspect(rules1_fp) #Sort Rules sort.rule1 <- sort(rules1_fp, by="lift") inspect(sort.rule1) #Visualization plot(sort.rule1[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue")) plot(sort.rule1) plot(sort.rule1, method="grouped", control=list(col=2))</pre>					
Hasil					
Rules					
<pre>> rules1_fp set of 416 rules > inspect(sort.rule1[1:5])</pre>					
	lhs	rhs	support	confidence	lift
[1]	{capital.gain=None, capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label<=50K}	0.2067974	0.9888901	1.314932
[2]	{capital.gain=None, hours.per.week=work-Holic, capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label<=50K}	0.2067974	0.9888901	1.314932
[3]	{capital.gain=None, marital.status=Never-married, age=YoungAdult}	=> {class.label<=50K}	0.2117316	0.9875129	1.313100
[4]	{capital.gain=None, hours.per.week=work-Holic, marital.status=Never-married, age=YoungAdult}	=> {class.label<=50K}	0.2117316	0.9875129	1.313100
[5]	{capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label<=50K}	0.2124397	0.9834067	1.307640

Aturan yang dihasilkan dengan skenario ini adalah berjumlah 416 dengan aturan dengan nilai lift tertinggi yaitu jika capital.gain=None, capital.loss=None, marital.status=Never-married, dan age=YoungAdult maka incomenya adalah kurang dari 50K. Selain itu, ada 415 aturan asosiasi lain yang dihasilkan.



Graph (1-10)



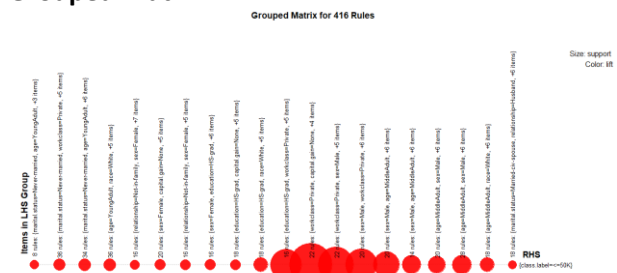
Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

Scatter Plot



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Grouped Matrix



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan $\leq 50K$. Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel $\leq 50K$.



Skenario 2

Skenario 2 menggunakan nilai support minimum 0,3 dan confidence 0,6 serta maxlength = 8.

Syntax

```
#rules2=====
rules2_fp <- rCBA::fpgrowth(dataset, support = 0.3,
                           confidence = 0.6, maxLength = 8,
                           consequent = "class.label", parallel=FALSE)

rules2_fp
inspect(rules2_fp)
#Sort Rules
sort.rule2 <- sort(rules2_fp, by="lift")
inspect(sort.rule2)
#Visualization
library(grid)
library(arulesviz)
plot(sort.rule2[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue"))
plot(sort.rule2)
plot(sort.rule2, method="grouped", control=list(col=2))
```

Hasil

Rules

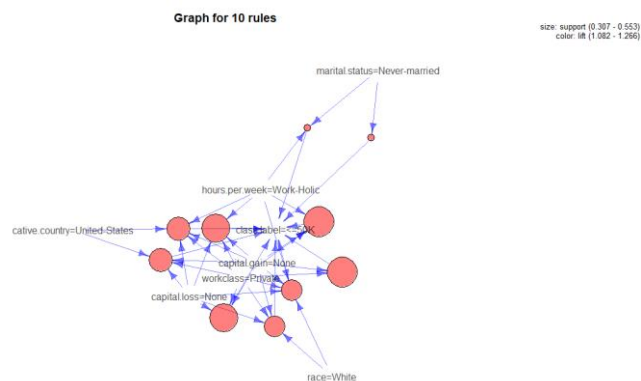
```
> rules2_fp
set of 118 rules

> inspect(sort.rule2[1:5])
```

lhs	rhs	support	confidence	lift
[1] {marital.status=Never-married}	=> {class.label<=50K}	0.3070983	0.9519204	1.265773
[2] {marital.status=Never-married, hours.per.week=work-Holic}	=> {class.label<=50K}	0.3070983	0.9519204	1.265773
[3] {capital.gain=None, capital.loss=None, workclass=Private}	=> {class.label<=50K}	0.5367748	0.8280370	1.101045
[4] {hours.per.week=work-Holic, capital.gain=None, capital.loss=None, workclass=Private}	=> {class.label<=50K}	0.5367748	0.8280370	1.101045
[5] {capital.gain=None, capital.loss=None, workclass=Private, cative.country=United-States}	=> {class.label<=50K}	0.4811037	0.8227570	1.094024

Aturan yang dihasilkan dengan skenario ini adalah berjumlah 118 dengan aturan dengan nilai lift tertinggi yaitu jika marital-status=Never Married pendapatannya dari 50K. Selain itu, ada 117 aturan asosiasi lain yang dihasilkan.

Graph (1-10)



Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

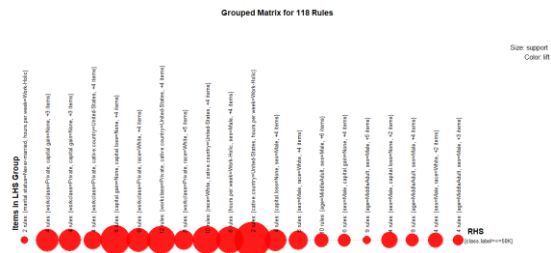


Scatter Plot



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Grouped Matrix



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan $\leq 50K$. Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel $\leq 50K$.

Skenario 3

Skenario 3 menggunakan nilai support minimum 0,2 dan confidence 0,7 serta maxlength = 8.

Syntax

```
#rules3 =====
rules3_fp <- rCBA::fpgrowth(dataset, support = 0.2,
                             confidence = 0.7, maxLength = 8,
                             consequent = "class.label", parallel=FALSE)

#Sort Rules
sort.rule3 <- sort(rules3_fp, by="lift")
inspect(sort.rule)

#Visualization
library(grid)
library(arulesviz)
plot(sort.rule3[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue"))
plot(sort.rule3)
plot(sort.rule3, method="grouped", control=list(col=2))
```

Hasil

Rules

```
> rules3_fp
set of 352 rules

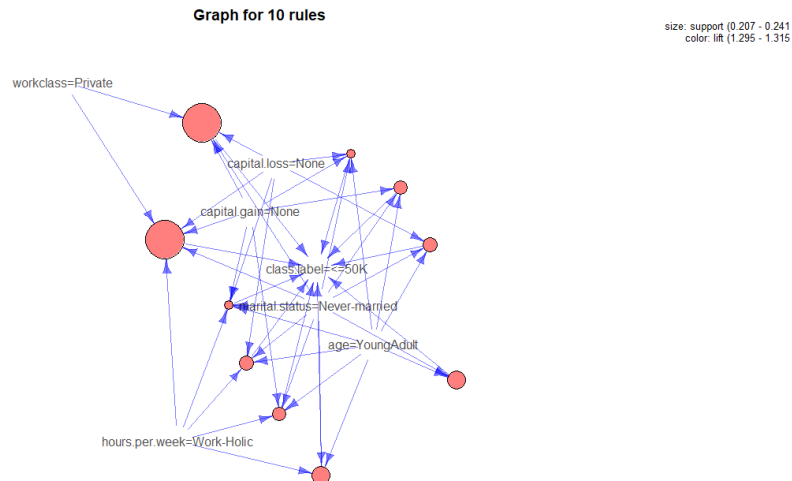
> inspect(sort.rule3[1:5])
```

	lhs	rhs	support	confidence	lift
[1]	{capital.gain=None, capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label= $\leq 50K$ }	0.2067974	0.9888901	1.314932
[2]	{capital.gain=None, hours.per.week=Work-Holic, capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label= $\leq 50K$ }	0.2067974	0.9888901	1.314932
[3]	{capital.gain=None, marital.status=Never-married, age=YoungAdult}	=> {class.label= $\leq 50K$ }	0.2117316	0.9875129	1.313100
[4]	{capital.gain=None, hours.per.week=Work-Holic, marital.status=Never-married, age=YoungAdult}	=> {class.label= $\leq 50K$ }	0.2117316	0.9875129	1.313100
[5]	{capital.loss=None, marital.status=Never-married, age=YoungAdult}	=> {class.label= $\leq 50K$ }	0.2124397	0.9834067	1.307640

Aturan yang dihasilkan dengan skenario ini adalah berjumlah 352 dengan aturan dengan nilai lift tertinggi yaitu jika capital.gain=None, capital.loss=None, marital.status=Never-married, dan age=YoungAdult maka incomenya adalah kurang dari 50K. Selain itu, ada 351 aturan asosiasi lain yang dihasilkan.

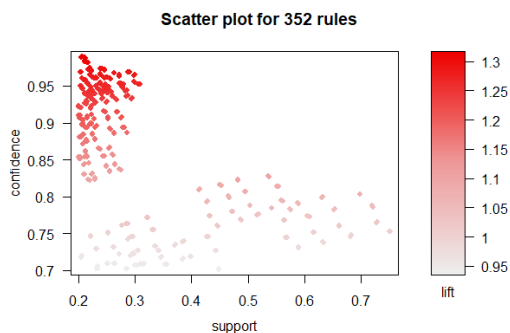


Graph (1-10)



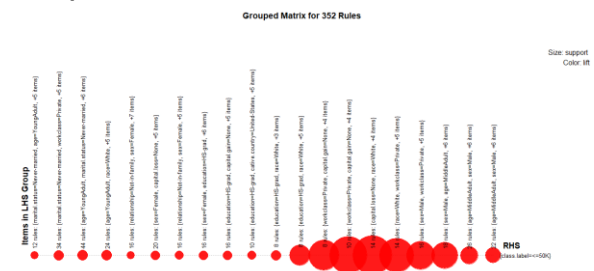
Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

Scatter Plot



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Grouped Matrix



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan $\leq 50K$. Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel $\leq 50K$.



Skenario 4

Skenario 4 menggunakan nilai support minimum 0,3 dan confidence 0,7 serta maxlength = 8.

Syntax

```
#rules4=====
rules4_fp <- rCBA::fpgrowth(dataset, support = 0.3,
                             confidence = 0.7, maxLength = 8,
                             consequent = "class.label", parallel=FALSE)

rules4_fp
inspect(rules4_fp)
#Sort Rules
sort.rule4 <- sort(rules4_fp, by="lift")
inspect(sort.rule4)
#Visualization
plot(sort.rule4[1:10], method="graph", control=list(nodeCol="red", edgeCol="blue"))
plot(sort.rule4)
plot(sort.rule4, method="grouped", control=list(col=2))
```

Hasil

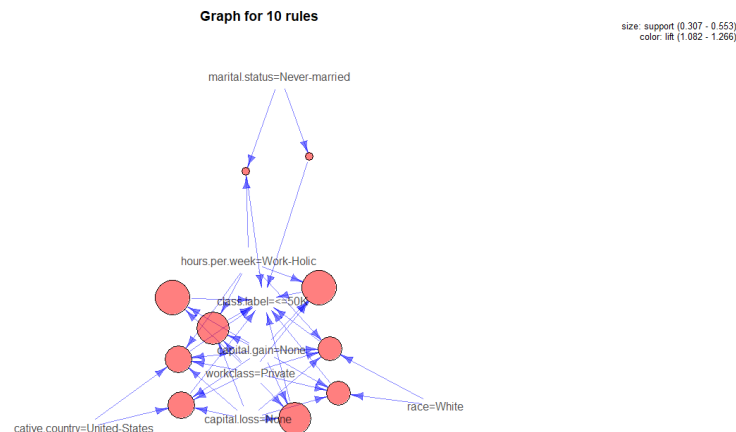
Rules

```
> rules4_fp
set of 100 rules

> inspect(rules4_fp[1:5])
  lhs                                     rhs      support confidence   lift
[1] {marital.status=Never-married} => {class.label=<=50K} 0.3070983 0.9519204 1.2657730
[2] {marital.status=Never-married,      => {class.label=<=50K} 0.3070983 0.9519204 1.2657730
    hours.per.week=work-Holic}
[3] {capital.gain=None,                  => {class.label=<=50K} 0.3154401 0.7086191 0.9422541
    age=MiddleAdult}
[4] {capital.gain=None,                  => {class.label=<=50K} 0.3041554 0.7277637 0.9677107
    age=MiddleAdult,
    capital.loss=None}
[5] {hours.per.week=work-Holic,          => {class.label=<=50K} 0.3041554 0.7277637 0.9677107
    capital.gain=None,
    age=MiddleAdult,
    capital.loss=None}
```

Aturan yang dihasilkan dengan skenario ini adalah berjumlah 110 dengan aturan dengan nilai lift tertinggi yaitu jika marital-status=Never Married pendapatannya dari 50K. Selain itu, ada 109 aturan asosiasi lain yang dihasilkan.

Graph (1-10)



Graph yang terbentuk dibatasi untuk 10 rules teratas saja agar graph dapat merepresentasikan data dengan baik (tidak tumpang tindih). Dari graph yang terbentuk, dapat dilihat aturan-aturan dengan nilai support yang tinggi dan nilai lift yang tinggi pula. Panah merepresentasikan hubungan, sedangkan besarnya lingkaran merepresentasikan nilai support. Semakin tinggi nilai support, semakin besar pula lingkaran yang terbentuk.

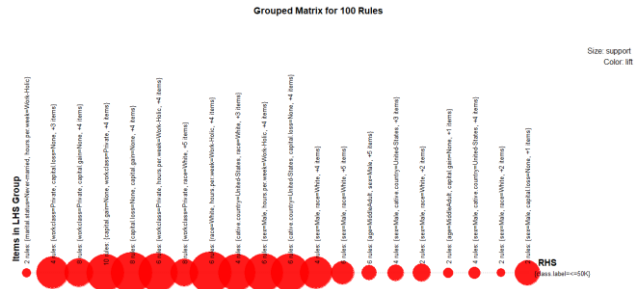


Scatter Plot



Dari scatter plot yang terbentuk, dapat dilihat persebaran nilai support, confidence dan lift dari aturan-aturan yang dihasilkan. Nilai lift direpresentasikan dengan kepekatan warna merah. Semakin tinggi nilai lift, maka semakin pekat warna merah pada representasi titik yang dihasilkan. Dapat diketahui dari scatter plot diatas bahwa aturan dengan nilai confidence tinggi dan lift yang tinggi mayoritas memiliki nilai support yang rendah. Sedangkan aturan dengan nilai lift yang kecil relatif memiliki nilai confidence yang kecil pula.

Grouped Matrix



Dari grouped matrix yang dihasilkan, dapat diketahui besaran support untuk grup LHS tertentu dan RHS pada aturan-aturan yang terbentuk. Ukuran dari lingkaran yang muncul bergantung pada nilai support, semakin kecil nilai support maka semakin kecil pula lingkaran yang dihasilkan. Pada grouped matrix yang terbentuk diketahui pula bahwa aturan-aturan yang dihasilkan mencakup hanya 1 jenis RHS yaitu pendapatan $\leq 50K$. Hal ini cukup masuk akal karena persebaran label kelas tidak simetris, dimana data mayoritas berlabel $\leq 50K$.

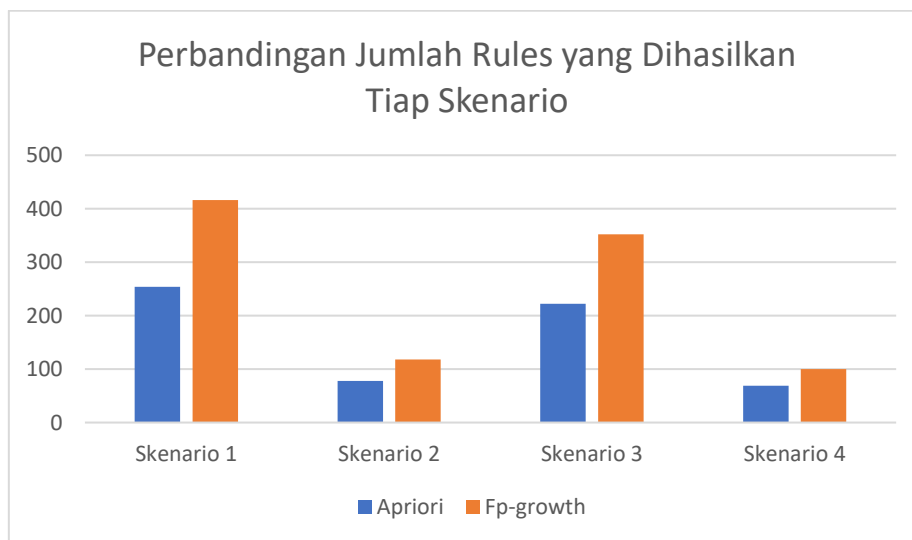


Jumlah Rules

Jumlah aturan-aturan (rules) yang dihasilkan berbeda dengan menggunakan algoritma serta skenario yang berbeda. Berikut merupakan jumlah aturan yang dihasilkan :

	Jumlah Rules yang dihasilkan	
	Apriori	Fp-Growth
Skenario 1 (min. sup = 0.2 , min. conf = 0.6)	254	416
Skenario 2 (min. sup = 0.3 , min. conf = 0.6)	78	118
Skenario 3 (min. sup = 0.2 , min. conf = 0.7)	222	352
Skenario 4 (min. sup = 0.3 , min. conf = 0.7)	69	100

Dari angka-angka tersebut, kemudian dibentuk barplot untuk memvisualisasikan data dengan lebih baik seperti berikut.



Dari hasil visualisasi diatas, dapat diketahui beberapa hal yakni :

- Jumlah aturan asosiasi yang dihasilkan pada algoritma apriori dengan menggunakan nilai min. support dan min. confidence yang sama selalu lebih kecil daripada jumlah aturan asosiasi yang dihasilkan dengan algoritma fp-growth.
- Besar min. support yang sama dengan menggunakan algoritma yang sama, namun min. confidence yang lebih besar akan menghasilkan jumlah aturan yang lebih kecil karena aturan-aturan yang tidak memenuhi min. confidence tersebut akan dipangkas, begitupula sebaliknya.
- Besar min. confidence yang sama dengan menggunakan algoritma yang sama, namun min. support yang lebih besar akan menghasilkan jumlah aturan yang lebih kecil karena aturan-aturan yang tidak memenuhi min. support tersebut akan dipangkas, begitupula sebaliknya.



Support & Confidence

Support dan confidence merupakan ukuran-ukuran yang digunakan untuk aturan-aturan yang dibangun. Support merepresentasikan persentase jumlah transaksi yang berisi X dan Y dari keseluruhan transaksi, dengan kata lain support menunjukkan seberapa sering kemunculan terjadi. Support dirumuskan sebagai berikut :

$$\text{Support, } s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N}$$

Sedangkan Confidence menunjukkan persentasi banyaknya Y pada transaksi yang mengandung X., dengan kata lain confidence menunjukkan seberapa dapat dipercayanya suatu aturan yang dibentuk. Confidence dirumuskan sebagai berikut :

$$\text{Confidence, } c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

Dari skenario dan algoritma yang telah dijalankan, confidence dan support yang dihasilkan ditunjukkan pada tabel dibawah.

Tabel 1. Perbandingan Range Support dan Confidence pada Masing-masing Algoritma

	Range Support		Range Confidence	
	Apriori	Fp-Growth	Apriori	Fp-Growth
Skenario 1 (min. sup = 0.2 , min. conf = 0.6)	0.2004248 - 0.7289906	0.2004248 - 0.7520467	<u>0.6024128 -</u> <u>0.9888901</u>	0.6024128 - 0.9888901
Skenario 2 (min. sup = 0.3 , min. conf = 0.6)	0.3013896 - 0.7289906	0.3013896 - 0.7520467	<u>0.6676055 -</u> <u>0.9519204</u>	0.6676055 - 0.9519204
Skenario 3 (min. sup = 0.2 , min. conf = 0.7)	0.2005576 - 0.7289906	0.2006904 - 0.7520467	<u>0.7019661 -</u> <u>0.9888901</u>	0.7019661 - 0.9888901
Skenario 4 (min. sup = 0.3 , min. conf = 0.7)	0.3013896 - 0.7289906	0.3013896 - 0.7520467	<u>0.7025108 -</u> <u>0.9519204</u>	0.7025108 - 0.9519204

Dari tabel diatas, didapatkan informasi mengenai perbandingan range support dan confidence pada algoritma Apriori dan Fp-Growth. Algoritma dijalankan pada 4 skenario berbeda dan menghasilkan range support serta confidence yang berbeda. Nilai range support tertinggi adalah 0.7520476 dan dihasilkan oleh algoritma Fp-growth. Meskipun begitu, perbedaan nilai range support antara algoritma Apriori dan Fp-Growth sebenarnya tidak jauh, yakni hanya berkisar 0.03.

Selain range support, didapatkan pula informasi mengenai range confidence dari masing-masing algoritma. Nilai range confidence tertinggi adalah 0.9888901. Menariknya adalah tidak adanya perbedaan range confidence antara algoritma Apriori dengan Fp-Growth. Meskipun tidak ada perbedaan dalam range confidence, tidak menutup kemungkinan bahwa persebaran confidence dari kedua algoritma tersebut tidak sama (dapat dilihat dengan mean, median, dan modus).



Interest Factor (Lift)

Salah satu evaluasi kemenarikan pada suatu aturan asosiasi adalah dengan menggunakan Interest Factor, atau yang juga bisa disebut dengan "lift". Interest Factor dapat digunakan untuk merepresentasikan tingkat kemenarikan dari sebuah pola bersifat *meaningful* yang telah dibangun.

$$I(A, B) = \frac{s(A, B)}{s(A) \times s(B)} = \frac{N f_{11}}{f_{1+} f_{+1}}.$$

Interest factor juga umumnya dapat menggambarkan keberagantungan suatu hubungan statistik antar variabel. Hubungan tersebut dapat dilihat pada nilai interest factor, sebagai berikut :

$$I(A, B) \begin{cases} = 1, & \text{if } A \text{ and } B \text{ are independent;} \\ > 1, & \text{if } A \text{ and } B \text{ are positively related;} \\ < 1, & \text{if } A \text{ and } B \text{ are negatively related.} \end{cases}$$

Pada studi kasus kali ini, nilai Lift yang dihasilkan adalah sebagai berikut.

	Range Lift	
	Apriori	Fp-Growth
Skenario 1 (min. sup = 0.2 , min. conf = 0.6)	0.8010310-1.314932	0.8010310 - 1.314932
Skenario 2 (min. sup = 0.3 , min. conf = 0.6)	0.8877181-1.265773	0.8877181 - 1.265773
Skenario 3 (min. sup = 0.2 , min. conf = 0.7)	0.9334076-1.314932	0.9334076 - 1.314932
Skenario 4 (min. sup = 0.3 , min. conf = 0.7)	0.9341319-1.265773	0.9341319 - 1.265773

Dari hasil yang didapatkan pada percobaan menggunakan algoritma apriori dan fp-growth dengan empat skenario, nilai minimum dan maximum (range) interest factor sama pada tiap skenario yang dijanakan. Hal ini berkaitan dengan nilai range confidence yang sama pula. Mengingat nilai lift adalah nilai rasio perbandingan antara confidence dengan expected confidence.

Dapat dilihat pula bahwa nilai lift paling tinggi yaitu 1.314932 dihasilkan pada skenario dengan nilai minimum support sebesar 0.2 baik dengan menggunakan algoritma apriori dan fp-growth. Nilai lift yang lebih dari 1 tersebut menyiratkan bahwa variabel-variabel pada aturan asosiasi tersebut berhubungan positif. Sedangkan nilai lift yang terendah yakni 0.8010310 dihasilkan pada skenario 1 yaitu dengan menggunakan minimum support sebesar 0,2 dan min. confidence sebesar 0,6. Nilai lift yang kurang dari 1 ini menyiratkan bahwa variabel-variabel pada aturan asosiasi tersebut berhubungan secara negatif.

Dari sini, dapat diketahui bahwa skenario 3 dengan min. support 0.2 dan min.confidence 0.7 dapat menghasilkan nilai lift yang relatif paling tinggi dibandingkan pada skenario lainnya yaitu 0.9334076-1.314932.



KESIMPULAN

Pada kasus analisis asosiasi ini terdapat 2 penerapan algoritma yaitu Apriori dan FPGrowth. Setiap algoritma yang diterapkan menggunakan skenario yang berbeda. Untuk pencarian frequent itemset terdapat 2 skenario dengan nilai min support yang berbeda yaitu skenario 1 = 0.2 dan skenario 2 = 0.3. Namun, pencarian rules atau aturan akan digunakan 4 skenario yaitu membandingkan antara min support 0.2 dan 0.3 dengan min confidence 0.6 dan 0.7. Dari hasil penerapan tersebut dapat disimpulkan bahwa:

1. Penambahan min support dalam pencarian frequent itemset akan memengaruhi jumlah frequent item set. Semakin besar nilai min support maka semakin sedikit frequent itemset yang dihasilkan.
 2. Penerapan 4 skenario dalam mencari rules dari 2 algoritma didapatkan:
 - a. Nilai support tertinggi pada skenario 3 dan 4 dengan nilai sebesar 0.3013896 - 0.7520467 dimana nilai tersebut dihasilkan dari algoritma FP-Growth.
 - b. Nilai confidence tertinggi pada skenario 1 dengan nilai sebesar 0.6024128 - 0.9888901 dan skenario 3 dengan nilai sebesar 0.7019661 - 0.9888901 dimana 2 algoritma menghasilkan nilai confidence yang sama nilainya.
 - c. Nilai lift tertinggi pada skenario 3 dengan min. support 0.2 dan min.confidence 0.7 yaitu 0.9334076-1.314932.
- Jadi dapat disimpulkan bahwa skenario 3 relatif memiliki hasil aturan asosiasi dengan nilai support, confidence dan lift paling tinggi dengan menggunakan min support 0.2 dan min confidence 0.7. Hal ini berlaku untuk kedua algoritma yaitu Apriori dan FPGrowth.

