PERTEMUAN 10

TEKNIK *ASSOCIATION RULE MINING* DALAM R

TUJUAN PRAKTIKUM

Mahasiswa akan dapat menggunakan teknik-teknik dasar *association rule mining* dengan *tools* R

TEORI PENUNJANG

Analisis asosiasi berguna untuk menemukan hubungan penting yang tersembunyi di antara set data yang sangat besar. Hubungan yang sudah terbuka direpresentasikan dalam bentuk aturan asosiasi (*association rule)* atau set aturan item yang sering muncul. Aturan asosiasi adalah pernyataan implikasi bentuk X → Y, dimana X dan Y adalah *itemset* yang lepas (*disjoint),* yang memenuhi persyaratan $X \cap Y = \{\}$. Kekuatan aturan asosiasi dapat diukur dengan *support* dan *confidence. Support* digunakan untuk menentukan seberapa banyak aturan dapat diterapkan pada set data, sedangkan *confidence* digunakan untuk menentukan seberapa sering item di dalam Y muncul dalam transaksi yang berisi X. Beberapa formula yang digunakan dalam pendekatan ini yaitu:

$$\begin{aligned} \text{support}(A \Rightarrow B) &= P(A \cup B) \\ \text{confidence}(A \Rightarrow B) &= P(B|A) \\ &= \frac{P(A \cup B)}{P(A)} \\ \text{lift}(A \Rightarrow B) &= \frac{\text{confidence}(A \Rightarrow B)}{P(B)} \\ &= \frac{P(A \cup B)}{P(A)P(B)} \end{aligned}$$

LAPORAN PENDAHULUAN

1. Apa yang anda ketahui tentang *Association Rule Minning*?
2. Sebutkan algoritme asosiasi yang anda ketahui!

MATERI PRAKTIKUM

Algoritme Apriori dengan *tools* R

Titanic dataset adalah data 4 dimensi dengan informasi nasib penumpang di Titanic yang diringkas menurut *social class, sex, age,* dan *survival*. Untuk membuat data ini cocok untuk *association rule mining*, kita perlu melakukan praproses data, di mana setiap baris mewakili 1 orang dengan langkah sebagai berikut:

```
> str(Titanic)
table [1:4, 1:2, 1:2, 1:2] 0 0 35 0 0 0 17 0 118 154 ...
 - attr(*, "dimnames")=List of 4
  ..$ Class   : chr [1:4] "1st" "2nd" "3rd" "Crew"
  ..$ Sex     : chr [1:2] "Male" "Female"
  ..$ Age     : chr [1:2] "Child" "Adult"
  ..$ Survived: chr [1:2] "No" "Yes"
> df <- as.data.frame(Titanic)
> head(df)
  Class    Sex   Age Survived Freq
1   1st   Male Child       No    0
2   2nd   Male Child       No    0
3   3rd   Male Child       No   35
4  Crew   Male Child       No    0
5   1st Female Child       No    0
6   2nd Female Child       No    0
> titanic.raw <- NULL
> for(i in 1:4) { titanic.raw <- cbind(titanic.raw,
rep(as.character(df[,i]), df$Freq))}
> titanic.raw <- as.data.frame(titanic.raw)
> names(titanic.raw) <- names(df)[1:4]
> dim(titanic.raw)
[1] 2201    4
> str(titanic.raw)
'data.frame':       2201 obs. of  4 variables:
 $ Class   : Factor w/ 4 levels "1st","2nd","3rd",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ Sex     : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 2 2 ...
 $ Age     : Factor w/ 2 levels "Adult","Child": 2 2 2 2 2 2 2 2 2 2 ...
 $ Survived: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
> head(titanic.raw)
  Class  Sex   Age Survived
1   3rd Male Child       No
2   3rd Male Child       No
3   3rd Male Child       No
4   3rd Male Child       No
5   3rd Male Child       No
6   3rd Male Child       No
> summary(titanic.raw)
  Class         Sex          Age       Survived
1st :325   Female: 470   Adult:2092   No :1490
2nd :285   Male  :1731   Child: 109   Yes: 711
3rd :706
Crew:885
```

Setelah data selesai di praproses, selanjutnya dilakukan tahapan *association rule mining* dengan menggunakan algoritme apriori:

```
> library(arules)
> # find association rules with default settings
> rules <- apriori(titanic.raw)
> rules
> quality(rules) <- quality(rules)
> inspect(rules)
```

```
    lhs                     rhs             support confidence      lift
1   {}                   => {Age=Adult}   0.9504771  0.9504771 1.0000000
2   {Class=2nd}          => {Age=Adult}   0.1185825  0.9157895 0.9635051
3   {Class=1st}          => {Age=Adult}   0.1449341  0.9815385 1.0326798
4   {Sex=Female}         => {Age=Adult}   0.1930940  0.9042553 0.9513700
5   {Class=3rd}          => {Age=Adult}   0.2848705  0.8881020 0.9343750
6   {Survived=Yes}       => {Age=Adult}   0.2971377  0.9198312 0.9677574
7   {Class=Crew}         => {Sex=Male}    0.3916402  0.9740113 1.2384742
8   {Class=Crew}         => {Age=Adult}   0.4020900  1.0000000 1.0521033
9   {Survived=No}        => {Sex=Male}    0.6197183  0.9154362 1.1639949
10  {Survived=No}        => {Age=Adult}   0.6533394  0.9651007 1.0153856
11  {Sex=Male}           => {Age=Adult}   0.7573830  0.9630272 1.0132040
12  {Sex=Female,
     Survived=Yes}       => {Age=Adult}   0.1435711  0.9186047 0.9664669
13  {Class=3rd,
     Sex=Male}           => {Survived=No} 0.1917310  0.8274510 1.2222950
14  {Class=3rd,
     Survived=No}        => {Age=Adult}   0.2162653  0.9015152 0.9484870
15  {Class=3rd,
     Sex=Male}           => {Age=Adult}   0.2099046  0.9058824 0.9530818
16  {Sex=Male,
     Survived=Yes}       => {Age=Adult}   0.1535666  0.9209809 0.9689670
17  {Class=Crew,
     Survived=No}        => {Sex=Male}    0.3044071  0.9955423 1.2658514
18  {Class=Crew,
     Survived=No}        => {Age=Adult}   0.3057701  1.0000000 1.0521033
19  {Class=Crew,
     Sex=Male}           => {Age=Adult}   0.3916402  1.0000000 1.0521033
```

 # rules with rhs containing "Survived" only

```
> rules <- apriori(titanic.raw, parameter = list(minlen=2,
supp=0.005, conf=0.8), appearance = list(rhs=c("Survived=No",
"Survived=Yes"), default="lhs"), control = list(verbose=F))
> rules.sorted <- sort(rules, by="lift")
> inspect(rules.sorted)
```

> # find redundant rules

```
> subset.matrix <- is.subset(rules.sorted, rules.sorted)
> subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
> redundant <- colSums(subset.matrix, na.rm=T) >= 1
> which(redundant)
```

> # remove redundant rules

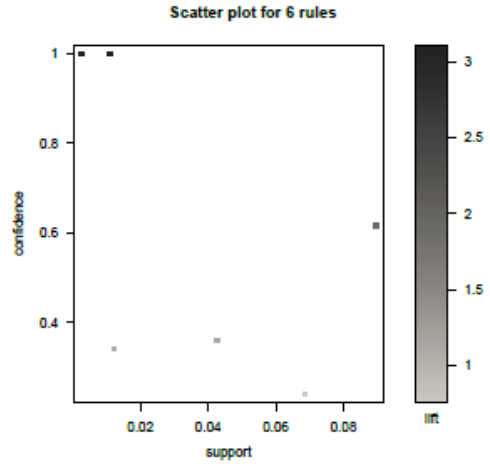
```
> rules.pruned <- rules.sorted[!redundant]
> inspect(rules.pruned)
```

> # interpretating rules

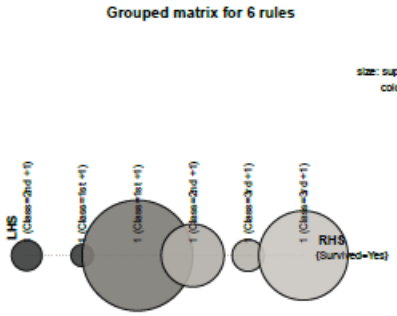
```
> rules <- apriori(titanic.raw, parameter = list(minlen=3,
supp=0.002, conf=0.2), appearance = list(rhs=c("Survived=Yes"),
lhs=c("Class=1st", "Class=2nd", "Class=3rd", "Age=Child",
"Age=Adult"), default="none"), control = list(verbose=F))
> rules.sorted <- sort(rules, by="confidence")
> inspect(rules.sorted)
```

>#Visualize rule

```
> library(arulesViz)
> plot(rules)
```

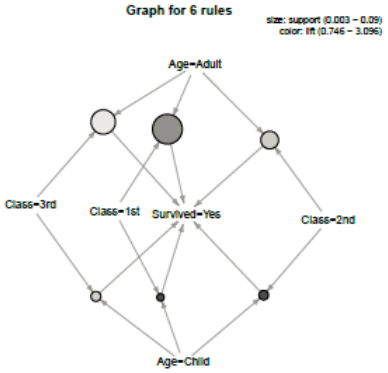



```
> plot(rules, method="grouped")
```




```
> plot(rules, method="graph")
```


```
> plot(rules, method="graph", control=list(type="items"))
```



DAFTAR PUSTAKA

Bouckaert, R. R.. et al. 2013. *WEKA Manual for Version 3-6-9*. Edition of January 21, 2013. http://jaist.dl.sourceforge.net/project/Weka/documentation/3.6.x/WekaManual-3-6-9.pdf . Accessed on 27 January 2013.

Han, J. (2006). Data Mining: Concepts and Technique. [Internet]. [diunduh 2014 Mar 8]. Tersedia pada: http://www.cs.uiuc.edu/homes/hanj/bk2/slidesindex.htm

Han J & Kamber M. 2011. *Data mining – Concept and Techniques.* Third Edition, Morgan-Kauffman, San Diego

Han, J., Pei, J., and Yin, Y. Mining *Frequent* Patterns without Candidate Generation. SIGMOD, 1-12. 2000.

R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large *database*s. Research Report RJ 9839, IBM Almaden Research Center, San Jose, California, June 1994.