Machine Learning - 1100-ML0ENG (Ćwiczenia informatyczne Z-23/24)

Home > My courses > Machine Learning - 1100-ML0ENG (Ćwiczenia informatyczne Z-23/24) > Association rules > Association rules I

Association rules I

```
install.packages("arules")
library(arules)
```

Transactions object

We begin with reading our transactions stored in the text file and create transactions.

```
d.tr<-read.transactions(file = "data.csv", format = "single",
 sep = ",", header = T,
 cols = c("order_id", "product_id"),
 rm.duplicates = FALSE,
 quote = "", skip = 0,
 encoding = "unknown")
d.tr@itemInfo
d.tr@data
```

Let's look at the parameters of **read.transactions()**, the function used to create the transactions object.

For the first parameter, file, we pass our text file where we have the transactions from the retailer.

The second parameter, format, can take any of two values, single or basket, depending on how the input data is organized. In our case, we have a tabular format with two columns--one column representing the unique identifier for our transaction and the other column for a unique identifier representing the product present in our transaction. This format is named single by arules. Refer to the arules documentation for a detailed description of all the parameters.

The most frequent items

The function **itemFrequency()** in the arules package takes a transaction object as input and produces the frequency count (the number of transactions containing this product) of the individual products:

```
# Item frequency
itemFrequency(d.tr, type = "absolute")
df1=as.data.frame(sort(itemFrequency(d.tr, type = "absolute"), decreasing = TRUE),
10) # Most frequent
```

```
df2=as.data.frame(sort(itemFrequency(d.tr, type = "absolute"), decreasing = FALSE),
10) # Least frequent
```

```
itemFrequencyPlot(d.tr,topN = 25)
```

Frequent itemsets

The apriori algorithm works in two phases. Finding the frequent itemsets is the first phase of the association rule mining algorithm. So we can also get the most common sets of our items.

```
support <- 0.01
# Frequent item sets
parameters = list(
 support = support,
 minlen = 2, # Minimal number of items per item set
 maxlen = 10, # Maximal number of items per item set
 target = "frequent itemsets"
)
```

```
freq.items <- apriori(d.tr, parameter = parameters)

freq.items.df <- data.frame(item_set = labels(freq.items)
 , support = freq.items@quality)
```

Association rules

```
rules.2 <- apriori(d.tr, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
summary(rules.2)
```

```


rules.2.sort <- sort (rules.2 , by="support", decreasing=TRUE)
inspect(rules.2.sort)
inspect(rules.2.sort[1:5])
write(rules.2.sort, file = "reguly_1.csv", quote=TRUE, sep = ",", col.names = NA)
```

To exclude certain items

Let us notice, that our itemsets consist mostly of the high frequency itemsets. However, we can ask the apriori method to ignore some of the items:

```
exclusion.items <- c('Banana','Bag of Organic Bananas')
```

```
#for the frequent items
```

```
freq.items2 <- apriori(d.tr, parameter = parameters,
 appearance = list(none = exclusion.items,
 default = "both"))
freq.items.df2 <- data.frame(item_set = labels(freq.items2)
 , support = freq.items2@quality)
```

```
#for the rules
```

```
rules.3 <- apriori(d.tr, parameter = list(supp = 0.001, conf = 0.8),
 appearance = list(none = exclusion.items))
rules.3.sort <- sort (rules.3 , by="support", decreasing=TRUE)
inspect(rules.3.sort[1:5])
```

We can create a vector of items to be excluded, and pass it as an appearance parameter to the apriori method. This will ensure that the items in our list are excluded from the generated itemsets.

Last modified: piątek, 18 listopada 2022, 6:14

Accessibility settings

Przetwarzanie danych osobowych

Platformą administruje Komisja ds. Doskonalenia Dydaktyki wraz z Centrum Informatyki Uniwersytetu Łódzkiego **Więcej**

Informacje na temat logowania

Na platformie jest wykorzystywana metoda logowania za pośrednictwem **Centralnego Systemu Logowania.**

Studentów i pracowników

Deklaracja dostępności

Uniwersytetu Łódzkiego obowiązuje nazwa użytkownika i hasło wykorzystywane podczas logowania się do systemu USOSweb.