Association Rules and Frequent Pattern Mining

Pedro Bastos e Juliana Fortes
DCC - FCUP
Data Mining II

April 4, 2016

1 Loading the data

Para carregar os dados para o R foi utilizado o seguinte código:

```
dia_22 <- read.csv2(file="/home/bastos/Documentos/DM2/dummydata/
22022016_0930_16.csv",
col.names=c("id","hora_e_loja","zona","hora_e_zona","tempo_per"))
dia_23 <- read.csv2(file="/home/bastos/Documentos/DM2/dummydata/
23022016_0930_16.csv",
col.names=c("id","hora_e_loja","zona","hora_e_zona","tempo_per"))
dia_24 <- read.csv2(file="/home/bastos/Documentos/DM2/dummydata/
24022016_0930_16.csv",
col.names=c("id","hora_e_loja","zona","hora_e_zona","tempo_per"))
dia_25 <- read.csv2(file="/home/bastos/Documentos/DM2/dummydata/
25022016_1030_17.csv",
col.names=c("id","hora_e_loja","zona","hora_e_zona","tempo_per"))
dia_26 <- read.csv2(file="/home/bastos/Documentos/DM2/dummydata/
26022016_1030_17.csv",
col.names=c("id","hora_e_loja","zona","hora_e_zona","tempo_per"))
```

2 Data exploration

O dataset representa as repetitivas entradas numa loja e consequente orientação dos clientes bem como o tempo de permanecia em determinada zona/sub-zona da loja. Cada entrada tem um id, id esse, que se refere ao dispositivo móvel de um cliente que entrou na loja. O dataset é constituído por as seguintes colunas: id do dispositivo, hora de entrada na loja, nome da zona, hora de entrada na zona e tempo de permanência em segundos.

To start with, we should have an idea of the dataset. Before that, we change the names of the columns to something more readable.

So, we have records, users and articles in the dataset. Notice that these are different users and different articles.

2.1 The Zipf distribution

To have an idea of the distribution of the data we will depict a frequency line sorted by size.

Here's one for the articles.

And another for the users.

We can observe a very typical Zipf distribution or "power law" both for articles and for users. This means that few articles (users) have very high frequencies and most have very low frequency. In other words there is a "long tail" in the distribution. This makes this kind of data hard for some predictive tasks since most articles (users) have a very low expression.

We can here also ask questions like: "Which are the most read articles?" or "Which are the most active users?". Answers can be given as lists, tables or barplots.

3 Geração das regras de associação

3.1 Regras para Zonas

Uma vez que algumas zonas são constituídas por varias sub-zonas. Decidimos ignorar as sub-zonas interpretando a passagem numa sub-zona como a passagem pela sua zona correspondente. Por exemplo, a sub-zona `Caracole 12` corresponde a zona `Caracole`.
Como a intenção comercial de um cliente nem sempre é a mesma, decidimos interpretar um cliente como um cliente diferente consoante o dia. Imaginemos, como exemplo, que o cliente com o $\widehat{\text{id}}$ 3 aparece no primeiro e no segundo dia, o cliente terá o $\widehat{\text{id}}$ 3 no primeiro dia e um $\widehat{\text{id}}$ diferente no segundo.

```
#dia_22
data <- data.frame(dia_22$id, gsub("\\s[0-9]*$", "", dia_22$zona))
colnames(data) <- c("id","zona")

#dia_23
maximo<-max(data$id)
new <- data.frame(dia_23$id + maximo, gsub("\\s[0-9]*$", "", dia_23$zona))
colnames(new) <- c("id","zona")
data <- rbind(data,new)

#dia_24
maximo<-max(data$id)
new <- data.frame(dia_24$id + maximo, gsub("\\s[0-9]*$", "", dia_24$zona))
colnames(new) <- c("id","zona")
data <- rbind(data,new)

#dia_25
maximo<-max(data$id)
new <- data.frame(dia_25$id + maximo, gsub("\\s[0-9]*$", "", dia_25$zona))
colnames(new) <- c("id","zona")
data <- rbind(data,new)
```

```
#dia_26
maximo<-max(data$id)
new <- data.frame(dia_26$id + maximo, gsub("\\s[0-9]*$", "", dia_26$zona))
colnames(new) <- c("id","zona")
data <- rbind(data,new)
```

O máximo garante apenas que nenhum îd se repete nos restantes dias. elimina o numero da sub-zona.
Depois disto é preciso garantir que a zonas são únicas por îd:

```
#valores unicos
data <- unique(data[c("id", "zona")])
```

O *dataset* **data** é composto 9267 entradas. Posto isto podemos gerar as regras:

```
#regras
library(carenR)
rls<-caren(data,Bas=TRUE)
```

Obtivemos 503 regras. Falta então definir o suporte mínimo, a confiança e o *improvement*. Começamos por definir o *improvement* como 0.05 para que não fossem geradas regras sem interesse. Um suporte mínimo de 0.04=(385/9267) e uma confiança de 0.75.

```
rls<-caren(data,Bas=TRUE, min.sup = 0.04, min.conf = 0.75, imp = 0.05, chi = T)
```

Com isto obtivemos um conjunto de 8 regras. Ordenamos finalmente os resultados por *Lift* pois este representa a importância do antecessor para a ocorrência do consequente.
Benhardt
Quem visita *Van Thiel Fendi Home* e *Caracole* também visita *Benhardt*.
Quem visita *Eichholtz* e *Fendi Home* também visita *Benhardt*.
Fendi Home
Quem visita *Eichholtz* e *Benhardt* também visita *Fendi Home*.
Quem visita *Eichholtz* e *Caracole* também visita *Fendi Home*.
Caracole
Quem visita *Van Thiel* e *Benhardt* também visita *Caracole*.
Quem visita Van *Van Thiel* e *Fendi Home* também visita *Caracole*.
Quem visita Van *Benhardt* e *Fendi Home* também visita *Caracole*.
Quem visita Van *Eichholtz* e *Fendi Home* também visita *Caracole*.

3.2 Regras para sub-Zonas

Para fazer o estudo do comportamento do cliente dentro de determina zona recorremos ao algoritmo sequencial *GSP*. Para isso selecionamos todas as zonas que tivessem mais de 2 sub-zonas pois a sequencia natural para uma zona de 2 sub-zonas, será, 1->2 e 2->1, sendo 1 a primeira sub-zona e 2 a segunda sub-zona. Interpretando o cliente/dia da mesma forma que se interpretou nas regras para zonas.

```
#dia_22
data_gsp_aux <- data.frame(dia_22$id[grepl("\\s[0-9]*$",dia_22$zona)],dia_22$zo
colnames(data_gsp_aux) <- c("id","zona","tempo")
```

```
#dia_23
new <- NULL
maximo<-max(data_gsp_aux$id)
new <- data.frame(dia_23$id[grepl("\\s[0-9]*$",dia_23$zona)] + maximo,dia_23$zo
colnames(new) <- c("id","zona","tempo")
data_gsp_aux <- rbind(data_gsp_aux,new)

#igual para os restantes dias
```

Sendo assim selecionamos todas as sub-zonas, o id e a hora de mudança de zona. E procedemos a eliminação das zonas com menos de 2 sub-zonas. Ficamos apenas com as seguintes zonas: *Benhardt*, *Caracole*, *Kitchen Accessories*, *Van Thiel*, *Fendi Home* e *Textile & Beding*. Obtemos um *dataset* com 16683 observações. Procedemos então a ordenação crescente por id, zona, e hora de mudança de zona.

```
library(chron)
data_gsp_aux <- data_gsp_aux[order(data_gsp_aux$id,strhead(as.character(data_gs
```

Posto isto passamos os *datasets* referentes a cada zona ao *weka* onde existe uma implementação do GSP. Como é óbvio só terão interesse as regras em que o antecedente não é igual ao precedente. Obtemos as seguintes regras de duas sequencias:

```
Caracole (6809 Observa  es)

[10] <{Caracole 8}{Caracole 12}> (103)
[19] <{Caracole 2}{Caracole 11}> (111)
[31] <{Caracole 6}{Caracole 11}> (109)
[38] <{Caracole 3}{Caracole 11}> (97)
[50] <{Caracole 1}{Caracole 11}> (129)
[63] <{Caracole 7}{Caracole 12}> (95)
[72] <{Caracole 4}{Caracole 11}> (120)
[78] <{Caracole 11}{Caracole 12}> (81)
[90] <{Caracole 5}{Caracole 12}> (120)
[98] <{Caracole 9}{Caracole 11}> (106)
[103] <{Caracole 12}{Caracole 11}> (81)
[107] <{Caracole 10}{Caracole 12}> (74)

Benhart (3067 Observa  es)

[9] <{Benhardt 7}{Benhardt 4}> (44)
[8] <{Benhardt 4}{Benhardt 6}> (85)
[12] <{Benhardt 1}{Benhardt 4}> (73)
[21] <{Benhardt 3}{Benhardt 6}> (62)
[28] <{Benhardt 6}{Benhardt 2}> (72)
[33] <{Benhardt 2}{Benhardt 6}> (71)
[36] <{Benhardt 5}{Benhardt 4}> (76)

Van Thield (1507 Observa  es)
```

```
[2] <{Van Thiel 4}{Van Thiel 2}> (59)
[6] <{Van Thiel 3}{Van Thiel 2}> (64)
[10] <{Van Thiel 2}{Van Thiel 3}> (63)
[15] <{Van Thiel 1}{Van Thiel 3}> (62)
[19] <{Van Thiel 5}{Van Thiel 3}> (35)

Fendi Home (3174 Observa es)

[3] <{Fendi Home 2}{Fendi Home 3}> (125)
[8] <{Fendi Home 1}{Fendi Home 4}> (108)
[12] <{Fendi Home 3}{Fendi Home 4}> (120)
[15] <{Fendi Home 4}{Fendi Home 3}> (110)
[18] <{Fendi Home 5}{Fendi Home 4}> (56)

Kitchen Accessories (453 Observa es)

[2] <{Kitchen Accessories 3}{Kitchen Accessories 1}> (11)
[4] <{Kitchen Accessories 2}{Kitchen Accessories 10}> (13)
[7] <{Kitchen Accessories 1}{Kitchen Accessories 10}> (11)
[?] <{Kitchen Accessories 10}{Kitchen Accessories ?}> (?)

Textile & Bedding (1673 Observa es)

[3] <{Textile & Bedding 2}{Textile & Bedding 1}> (36)
[7] <{Textile & Bedding 4}{Textile & Bedding 3}> (49)
[8] <{Textile & Bedding 1}{Textile & Bedding 2}> (38)
[11] <{Textile & Bedding 3}{Textile & Bedding 4}> (41)
```

O número de observações da sequencia é apresentado dentro dos parênteses curvos.

Then, we load the `carenR` package and launch the caren association rule (AR) discovery with the `caren` command. We specify that the data set is in "Bas" format. The parameters min.sup, min.conf and imp control the number of rules generated.

The process generated rules. Which articles appear most as consequents? To answer that, we collect consequents in the rules data.frame and select the most frequent ones.

Are these the same as the most frequent ones? It is likely that there is some overlap. However, interesting articles may appear frequently as a rule consequent without necessarily being frequent overall. We can actually plot the relation between the two frequencies.