# 1 Topic

## Mining frequent itemsets with the FREQUENT ITEMSETS component.

Searching regularities from dataset is the main goal of the data mining. They may have various representations. In the market basket analysis, we search the co occurrences of goods (items) i.e. the goods which are often purchased simultaneously. They are called "frequent itemset". For instance, one result may be "milk and bread are purchased simultaneously in 10% of caddies".

Frequent itemset mining is often presented as the preceding step of the association rule learning algorithm. At the end of the process, we highlight the direction of the relation. We obtain rules. For instance, a rule may be "90% of the customers which buy milk and bread will purchase butter also". This kind of rule can be used in various manners. For instance, we can promote the sales of milk and bread in order to increase the sales of butter.

In fact, frequent itemsets provide also valuable information. Detecting the goods which are purchased simultaneously enables to understand the relation between them. It is a kind of variant of the clustering analysis. We search the items which come together. For instance, we can use this kind of information in order to reorganize the shelves of the store.

In this tutorial, we describe the use of the FREQUENT ITEMSETS component under Tanagra. It is based on the Borgelt's "apriori.exe" program. We use a very small dataset. It enables to everyone to reproduce manually the calculations. But, in a first time, we describe some definitions about the frequent itemset mining process.

## 2 Frequent itemset mining

<b>S1</b>	<b>S2</b>	<b>S3</b>	<b>S4</b>
1	0	1	0
0	1	0	0
0	0	0	1
0	1	1	1
0	1	1	0
0	1	1	0
1	1	1	1
1	0	1	0
1	1	1	0
1	1	1	0

Our data file contains 10 instances (transactions) and 4 items<sup>1</sup>.

Each row corresponds to a customer of an insurance company. Each column is an insurance contract. E.g. the customer n°1 has the contracts S1 and S3, etc. The objective is to identify products (contracts) that are placed together.

Item. An item corresponds to a product. We have 4 items (S1, S2, S3 and S4) here.

**Support**. The support of an item corresponds to the number of transactions in which it appears. For instance, the support of  $\{S_1\}$  is 5. The support can be also computed as a percentage. In this case, we divide the absolute support by the number of the transactions e.g.  $SUP(\{S_1\}) = 5/10 = 50\%$ .

<sup>&</sup>lt;sup>1</sup> <u>http://www.dataminingarticles.com/closed-maximal-itemsets.html</u>

**Itemset**. An itemset is a set of items, e.g.  $\{S_1, S_2\}$  is an itemset with the cardinal CARD( $\{S_1, S_2\} = 2$ . The support of an itemset counts the number of transactions in which the items appear simultaneously e.g. SUP( $\{S_1, S_2\}$ ) =  $_3/_{10}$  =  $_30\%$ . An itemset may be a singleton i.e. an itemset with a cardinal equal to 1.

**Frequent itemset**. An itemset is called frequent if its support is higher than a user-specified value. This last one is the main parameter of the algorithm. For instance, if we set the minimum support to 2 (or 20%) for our dataset, we obtain the following frequent itemsets (all non-grayed itemset into the graph).



Figure 1 – Mining the frequent itemsets

**Superset**. A superset is an itemset defined in relation to another itemset. For instance,  $B = \{S_1, S_2, S_3\}$  is a superset of  $A = \{S_1, S_2\}$  because CARD (A) < CARD (B) and  $A \subset B$  i.e. all the items present into A are also present into B. If A is not frequent, then B is not frequent also. This property enables to decrease considerably the computation time during the extraction of frequent itemsets from a database.

**Closed itemset**. A frequent itemset is called "closed" if all its supersets have a support lower than its own support. For instance  $\{S_1, S_3\}$  [SUP ( $\{S_1, S_3\}$ ) = 5/10] is a closed frequent itemset because none of its supersets has the same support: SUP ( $\{S_1, S_2, S_3\}$ ) = 3/10, SUP( $\{S_1, S_3, S_4\}$ ) = 1/10.

**Maximal itemset**. A frequent itemset is called "maximal" if none of its supersets is frequent. For instance,  $\{S_1, S_2, S_3\}$  is maximal because it has only one superset  $\{S_1, S_2, S_3, S_4\}$  and this last one is not frequent (SUP ( $\{S_1, S_2, S_3, S_4\}$ ) = 1/10 < 2/10).

**Generator itemset**. A is a generator itemset if it does not exist an itemset B such as  $B \subset A$  and SUP(B) = SUP(A). In other words, an itemset is generator if all its sub-itemsets have a support strictly higher than its own support. For instance, {S1, S2, S3} (support = 4/10) is not generator because {S1,

S2}, one of its sub-itemsets, have the same support (4/10). Whereas {S2, S4} (support = 2/10) is generator because SUP ({S2}) = 7/10 and SUP ({S4}) = 3/10.

## 3 Itemset mining with Tanagra

We load the «<u>itemset\_mining.xls</u> » data file into Excel. We select the data range; then we click on the TANAGRA / EXECUTE TANAGRA menu which is installed with the "tanagra.xla" add-in<sup>2</sup>.

C		· (°" • ) :	⇒ iter	nset_minin	ng.xls [Mod	e de comp	oatibilité] -	Microsoft Exe	cel	l	(		×
	Accueil	Insertion	Mise en pa	Formules	Données	Révision	Affichage	Développe	Compléme	Acrobat	0.		х
	Sipina 🔻												
	Tanagra 🔻		_										
Í	Execute	Tanagra 1											
Cor	About												
Ľ	D11	•	Ex	ecute Tana	igra	$\mathbf{v}$				l	X		≯
	А	В					61						
1	S1	S2	2			luaing the i	name of the	attributes fi	rst row):				
2	1	0		\$As	\$1:\$D\$11					_			
3	0	1						ок	Cance				
4	0	0						UK	Cance	-			=
5	0	1											
6	0	1		1	0								
7	0	1		1	0								
8	1	1		1	1								
9	1	0		1	0								
10	1	1		1	0								
11	1	1		1	0								
12													-
II I	→ → data	set So	urce 🖉 🔁	7			14					•	T
Prêt	: 🛅		Moyenn	e : 0.575	Nb (non vid	les) : 44	Somme : 23		100 % 🧲	)(	<b>)</b> —	-(+	) .::

Tanagra is automatically started and the dataset is loaded.

💇 TANAGRA 1.4.41 - [Datase	et (tanC84D.txt)]				
💇 File Diagram Compo	nent Window	Help			_ 8 ×
🗅 📽 🔚   👫					
Analysis		Data	aset desc	ription	
Dataset (tanC84D.tx	4 attrit	oute(s) mple(s)		]	
		Attrib	ute Category	Informations	s
		51	Continue	-	<b>-</b>
		52	Continue	-	E
		53	Continue	-	
		54	Continue	-	
		C	omponents		
Data visualization	Statist	ics	Nonparame	tric statistic	instance selection
Feature construction	Feature se	lection	Reg	ression	Factorial analysis
PLS	Cluster	ing	Spv I	earning	Meta-spv learning
Spv learning assessment	Scorir	ng	Asso	ciation	
Correlation scatterplot	Scatterple	ot	🔛 View	dataset	
Export dataset	🖉 Scatterple	ot with labe	l View	multiple sca	tterplot

<sup>&</sup>lt;sup>2</sup> <u>http://data-mining-tutorials.blogspot.fr/2010/08/tanagra-add-in-for-office-2007-and.html</u> for Excel; <u>http://data-mining-tutorials.blogspot.fr/2011/07/tanagra-add-on-for-openoffice-calc-33.html</u> for Open Office.

We select the variables for the analysis using the DEFINE STATUS component. We set all the variables as INPUT ones.

TANAGRA 1.4.41 - [Dataset (tanCa	84D.txt)]	
🏆 File Diagram Component A	Hindow Help - B	×
D 📽 🔲 🗱		
Analysis	Define attribute statuses	~
⊡		
Define status 1	Parameters	
	Attributes :	
	C S1 S1	
	C S2 S2	=
	C         S2           C         S3           C         S4	
		-
Data visualization		
Feature construction F		
PLS	B     B     Clear all     Clear selected	
Spv learning assessment	E E Clear all Clear selected	
Correlation scatterplot		
Export dataset	OK Cancel Help	
		4

## 3.1 Frequent itemsets

To extract the frequent itemsets, we add the component into the diagram. We click on the PARAMETERS contextual menu to specify the settings of the analysis.

TANAGRA 1.4.41 - [Datas	et (tanC84D.txt)]		
Tile Diagram Compo	onent Window Help		_ 8 ×
D 📽 🖫   🗱			$\frown$
Analysis	Dat	aset des	equent Itemsets
⊡ 🛄 Dataset (tanC84D.t	xt) 4 attr	ibute(s)	
🖻 🎎 Define status 1	10 ex	cample(s)	Parameters
Frequent Ite	Parameters	egory	Min Support: 0.2
	Execute	itinue	Max support : 1
	View	itinue	Min length : 2
	54	Continue	Max length : 4
			Itemset type
	1	Components	Frequent
Data visualization	Statistics	Nonparam	© Closed
Feature construction	Feature selection	Reg	© Maximal
PLS	Clustering	Spv	Generators
Spv learning assessment	Scoring	Ass	Rule base : outfreg.rul
🕄 A priori 🕄	A priori PT 🛛 🚦 Freq	quent Itemsets	
🕄 A priori MR	Assoc Outlier 💦 Spv .	Assoc Rule	<u></u>
			OK Cancel Help
			<b>ت</b> ثنہ

We set the minimal support to 20%. We do not modify the other parameters. By default, the tool extracts the frequent itemsets of cardinal between MIN LENGTH = 2 and MAX LENGTH = 4. So, the itemsets composed of a singleton are not generated.

TANAGRA 1.4.41 - [Frequ	ent Itemsets 1]				
💇 File Diagram Compo	onent Window H	lelp			_ 8 ×
D 📽 🔲 👪					
Analysis				Execution log	
Dataset (tanC84D.b		version 5.57 (2011. reading C:\Users\M [0.00s]. filtering, sorting and	09.02) (c) 1996 aison\AppData\Loc recoding items g transactions [6 tree [8 node(s)]	[4 item(s)] done [0.00s]. /10 transaction(s)] done [0.00 done [0.00s].	tem(s), 10 transaction(s)] done
			ITEM	SETS [#7 itemsets loaded]	
		N* Description			
		1 S4 /\ S2 /\ S3			
		2 S4 /\ S2			
		3 S4 /\ S3			
		4 S1 /\ S3	<b>F</b>		
		5 S1 /\ S2 /\ S3			
		6 S1 /\ S2			
		7 S2 /\ S3			
		•		III	•
		C	omponents		
Data visualization	Statistics	s Nonpara	metric statistics	Instance selection	Feature construction
Feature selection	Regressio	n Fact	orial analysis	PLS	Clustering
Spv learning	Meta-spv lear	rning Spv lear	ning assessment	Scoring	Association
A priori 8.	A priori MR	A priori PT	Assoc Outlie	er 🍀 Frequent Itemse	ets 🏶 Spv Assoc Rule 🗄
					, 

We click on the VIEW menu to start the analysis. The "apriori.exe" program is launched. The extracted itemsets are displayed into the visualization window.

## 3.2 Closed itemsets

To obtain the closed itemsets, we modify the settings by clicking again on the PARAMETERS menu. We set the following values (especially **ITEMSET TYPE = CLOSED**).

Frequent Itemsets
Parameters
Min Support : 0.2
Max support : 1
Min length : 2
Max length : 4
□ Itemset type ○ Frequent
Closed     Maximal
© Generators Rule base : outfreq.rul
OK Cancel Help

We validate our choice. We click on the VIEW menu, we obtain 4 closed itemsets.

TANAGRA 1.4.41 - [Freque	ent Itemsets 1]					
💇 File Diagram Compo	onent Window H	lelp			_ 8 ×	
D 📽 🔚 🔛						
Analysis				Execution log		
⊡-∰ Dataset (tanC84D.tx ⊡-∰ Define status 1 □-∰ Frequent Ite	, 	version 5 reading ( [0.00s]. filtering, s sorting a building t		2011 Christian Borgelt al\Temp\dat7B97.tmp [4 itr [4 item(s)] done [0.00s]. /10 transaction(s)] done [0.00 done [0.00s].	em(s), 10 transaction(s)] done	
			ITEM	SETS [#4 itemsets loaded]		
		1 S4 2 S1 3 S1	rscription ∧ S2 ∧ S3 ∧ S2 ∧ S3 ∧ S3 ∧ S3	m		
			Components			
Data visualization	Statistics	5	Nonparametric statistics	Instance selection	Feature construction	
Feature selection	Regressio	ression Factorial analysis PLS Clustering				
Spv learning	Meta-spv lear	rning	Spv learning assessment	Scoring	Association	
SA priori	A priori MR	8: A prior	i PT Assoc Outlie	er 🕄 Frequent Itemse	ts Spv Assoc Rule +	

## 3.3 Maximal itemsets

We set new parameters (**ITEMSET TYPE = MAXIMAL**) into the dialog settings. We obtain 2 maximal itemsets.

version 5.57 (2011.09.02) (c) 1996-2011 Christian Borgel	ť					
Analysis         Execution log           Image: State (tanC84D.txt)         D:\Temp\Exe\exe\apriori.exe - find frequent item sets with the aversion 5.57 (2011.09.02)           Image: State state 1         Image: State s	ť					
□ □ □ Dataset (tanC84D.bxt) □ □ ↓ 2 Define status 1 □.\Temp\Exe\exe\approxi.exe - find frequent item sets with the item sets witem sets with the	ť					
ersion 5.57 (2011.09.02) (c) 1996-2011 Christian Borgel	ť					
[0.00s]. filtering, sorting and recoding items [4 item(s)] done [0.00s].	reading C:\Users\Maison\AppData\Local\Temp\dat396A.tmp [4 item(s), 10 transaction(s)] done [0.00s]. filtering, sorting and recoding items [4 item(s)] done [0.00s]. sorting and reducing transactions [6/10 transaction(s)] done [0.00s]. building transaction tree [8 node(s)] done [0.00s].					
ITEMSETS [#2 itemsets load           N*         Description           1         S4 \lambda S2 \lambda S3           2         IS1 \lambda S2 \lambda S3           4         III	ed]					
Components						
Data visualization Statistics Nonparametric statistics Instance selection	Feature construction					
Feature selection Regression Factorial analysis PLS	Clustering					
Spv learning Meta-spv learning Spv learning assessment Scoring	Association					
I Spv Assoc Tree I A priori MR I Assoc Outlier I Spv Assoc Rule						

#### 3.4 Generator Itemsets

Last, to obtain generator itemsets, we set (**ITEMSET TYPE = GENERATORS**) into the dialog settings. We obtain 4 itemsets.

File Diagram Comp	onent Window H	elp				_	8
) 📽 🖪 🔤							
Analysis					Execution log		
Dataset (tanC84D.t	D:\Temp\Exelexe\apriori.exe - find frequent item sets with the apriori algorithm version 5.57 (2011.09.02) (c) 1996-2011 Christian Borgelt reading C:\Users\Maison\AppData\Local\Temp\dat9F8D.tmp [4 item(s), 10 transaction(s)] done [0.00s]. filtering, sorting and recoding items [4 item(s)] done [0.00s]. sorting and reducing transactions [6/10 transaction(s)] done [0.00s]. building transaction tree [8 node(s)] done [0.00s].						
				ITEMSE	TS [#4 itemsets loaded]		
		1 S4 2 S4	escription /\ S2 /\ S3 /\ S2				-
		4 S2 / S3				•	
			Componer	nts			
Data visualization	Statistics		Nonparametric st	atistics	Instance selection	Feature construction	1
Feature selection	Regression		Factorial anal	ysis	PLS	Clustering	
Spv learning	Meta-spv learn	ning	Spv learning asse	ssment	Scoring	Association	
8 A priori 8	A priori MR	A prio	ri PT 🏼 🚺 As	soc Outlier	Frequent Itemse	ts 🕄 Spv Assoc Rule	÷
4							

# 4 Extracting itemsets with R (« arules » package)

The "apriori" procedure based on the "arules" package<sup>3</sup> uses also the Borgelt's program. We can thus obtain the same results as above if we set the same settings for the calculations.

## 4.1 Data file importation

We use the « read.xls » command of the « xlsReadWrite<sup>4</sup> » package to read the data file.

#clear la mémoire
rm(list=ls())
#data importation
library(xlsReadWrite) #loading the package
setwd("...") #modifying the directory
#the first row corresponds to the name of the variables
#the dataset is located in the first sheet
dataset <- read.xls(file="itemset\_mining.xls",colNames=T,sheet=1)
#printing the values
print(dataset)</pre>

R provides the following results.

<sup>&</sup>lt;sup>3</sup> <u>http://cran.r-project.org/web/packages/arules/index.html</u>

<sup>&</sup>lt;sup>4</sup> <u>http://cran.r-project.org/web/packages/xlsReadWrite/index.html</u>

R F	R Con	sole				3
> 0	data	set	: <-	read	<pre>lining/Databases_for_mining/dataset_for_soft_dev_an\$ d.xls(file="itemset_mining.xls",colNames=T,sheet=1)</pre>	Î
> 1				set)		
			S3			
1	_	_	1	-		
2	0	1	0	0		
3	0	0	0	1		
4	0	1	1	1		
5	0	1	1	0		
6	0	1	1	0	·	_
7	1	1	1	1		
8	1	0	1	0		Ξ
9	1	1	1	0		
10	1	1	1	0		-
						-
4						

#### 4.2 Extracting the frequent itemsets

To extract the frequent itemsets, we load the "arules" library; then, we set the parameters of the analysis; last, we launch calculations. We note that we must transform the data frame into a matrix with the as.matrix(...) command.

#loading the package
library(arules)
#set the settings
params <- list(supp = 0.2, minlen = 2, maxlen = 4, target="frequent itemsets")
#computing the rules
result <- apriori(as.matrix(dataset), parameter = params)
#printing the itemsets
inspect(result)

R shows the itemsets. We obtain the same results as Tanagra.

R Console	
<pre>apriori - find association rules with the apriori algorithm version 4.21 (2004.05.09) (c) 1996-2004 Christian Borge set item appearances[0 item(s)] done [0.003]. set transactions[4 item(s), 10 transaction(s)] done [0.003]. sorting and recoding items [4 item(s)] done [0.003]. creating transaction tree done [0.003]. checking subsets of size 1 2 3 done [0.003]. writing [7 set(s)] done [0.003]. creating 54 object done [0.003]. &gt; inspect(result) items support 1 {52,</pre>	lt
S4} 0.2	
2 {53, 54} 0.2	-
3 {S1,	=
S2} 0.3	
4 {S1, S3} 0.5	
S3} 0.5	-
. ↓	E. ₹

**Note**: We note that the package "arule" is based on the 4.21 version of "apriori.exe". The extraction of the generator itemsets is not available.

## 4.3 Extracting the other kinds of itemsets

The results being the same that those of TANAGRA, we describe only the parameters of calculations for each type of itemset in this sub-section.

#### #mining the closed itemsets

```
params <- list(supp = 0.2, minlen = 2, maxlen = 4, target="closed frequent itemsets")</pre>
```

result <- apriori(as.matrix(dataset), parameter = params)

inspect(result)

#mining the maximal itemsets

params <- list(supp = 0.2, minlen = 2, maxlen = 4, target="maximally frequent itemsets")

result <- apriori(as.matrix(dataset), parameter = params)

inspect(result)

## 5 Conclusion

We update the external library "apriori.exe" of Borgelt (version 5.57) in the Tanagra 1.4.41. It is definitely faster<sup>5</sup>. We added in this occasion a new component for the extraction of the frequent itemsets (FREQUENT ITEMSETS) which is also based on the same program.

<sup>&</sup>lt;sup>5</sup> <u>http://data-mining-tutorials.blogspot.fr/2011/09/priori-pt-updated.html</u>