

第5章 連関規則

5.5 Rによる連関規則の発見

03T4027N 佐々 知広

温泉の効能の分析

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	温泉名	リウマチ	皮膚病	婦人病	高血圧	動脈硬化	痛風	腰痛	神経痛	創傷	胃腸病	喘息	水虫	糖尿病	火傷	眼病	湿疹
2	黒湯温泉	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
3	鷹の湯温泉	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
4	稲住温泉	1	0	1	0	0	0	1	1	0	1	0	0	0	0	0	0
5	玉川温泉	1	1	1	0	1	0	0	1	0	1	1	0	0	0	0	0
6	澄川温泉	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0
7	後生掛温泉	1	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0
8	孫六温泉	1	0	1	0	0	0	0	1	1	0	0	0	0	1	0	1
9	蟹場温泉	1	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0
10	大釜温泉	1	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0
11	妙の湯温泉	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1
12	鶴の湯温泉	1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0
13	乳頭温泉	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
14	仁別温泉	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
15	貝の沢温泉	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
16	秋田温泉	1	1	0	0	0	0	0	1	1	0	0	0	0	0	1	0
17	船沢温泉	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
18	能代温泉	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
19	湯ノ平温泉	1	1	1	0	0	0	0	1	0	0	0	0	0	1	0	0
20	横手温泉	1	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0
21	滝ノ沢温泉	0	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0
22	長瀨温泉	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	雪沢温泉	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0
24	ふるさわ温	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
25	下内沢温泉	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
26	日景温泉	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
27	矢立温泉	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
28	安楽温泉	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
29	親川ノ湯	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
30	三六温泉	1	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0

```
R Console
> library(arules)
> 温泉データ <- read.csv("onsen.csv",header=TRUE, row.names=1)
> 温泉データ <- as.matrix(温泉データ)
> 温泉トランザクション <- as(温泉データ,"transactions")
> summary(温泉トランザクション)
transactions as itemMatrix in sparse format with
95 rows (elements/itemsets/transactions) and
28 columns (items) and a density of 0.1443609

most frequent items:
リュマチ 神経痛 婦人病 創傷 皮膚病 (Other)
79 55 42 33 23 152

element (itemset/transaction) length distribution:
sizes
0 1 2 3 4 5 6 7 9
1 2 7 21 28 29 3 3 1

Min. 1st Qu. Median Mean 3rd Qu. Max.
0.000 3.000 4.000 4.042 5.000 9.000

includes extended item information - examples:
labels
1 リュマチ
2 皮膚病
3 婦人病

includes extended transaction information - examples:
transactionID
1 黒湯温泉
2 鷹の湯温泉
3 稲住温泉
> |
```

Min	最小値
1st Qu	第一四分位数
Median	中央値
Mean	平均値
3rd Qu	第三四分位数
Max	最大値

*row.names=1 データフレームの行名をcsvファイルのデータの第一列目とする。
 *as.matrix() 連関規則では0, 1の2値データに関しては最初にマトリックス形式 にしておく。

```
R Console
> inspect(温泉トランザクション)
  items                                transactionID
1  {リウマチ,                          黒湯温泉
   皮膚病,
   婦人病,
   高血圧,
   動脈硬化}
2  {リウマチ,                          鷹の湯温泉
   痛風,
   腰痛,
   神経痛,
   創傷}
3  {リウマチ,
   婦人病,
   腰痛,
   神経痛,
   胃腸病}
4  {リウマチ,                          玉川温泉
   皮膚病,
   婦人病,
   動脈硬化,
   神経痛,
   胃腸病,
   喘息}
5  {リウマチ,                          澄川温泉
   婦人病,
   痛風,
   神経痛}
6  {リウマチ,                          後生掛温泉
   神経痛,
   創傷,
   糖尿病,
   火傷}
```

```
R Console
> itemFrequency(温泉トランザクション)
  リウマチ      皮膚病      婦人病      高血圧      動脈硬化      痛風      腰痛
0.83157895 0.24210526 0.44210526 0.17894737 0.20000000 0.10526316 0.09473684
  神経痛      創傷      胃腸病      喘息      水虫      糖尿病      火傷
0.57894737 0.34736842 0.11578947 0.02105263 0.01052632 0.14736842 0.08421053
  眼病      湿疹      泌尿器病      関節痛      筋肉痛      角化症      運動器障害
0.03157895 0.08421053 0.01052632 0.05263158 0.04210526 0.03157895 0.20000000
  神経麻痺      肝臓病      痔疾      アトピー      打ち身      冷え症      便秘
0.01052632 0.01052632 0.05263158 0.02105263 0.05263158 0.01052632 0.03157895
> itemFrequencyPlot(温泉トランザクション)
>
```

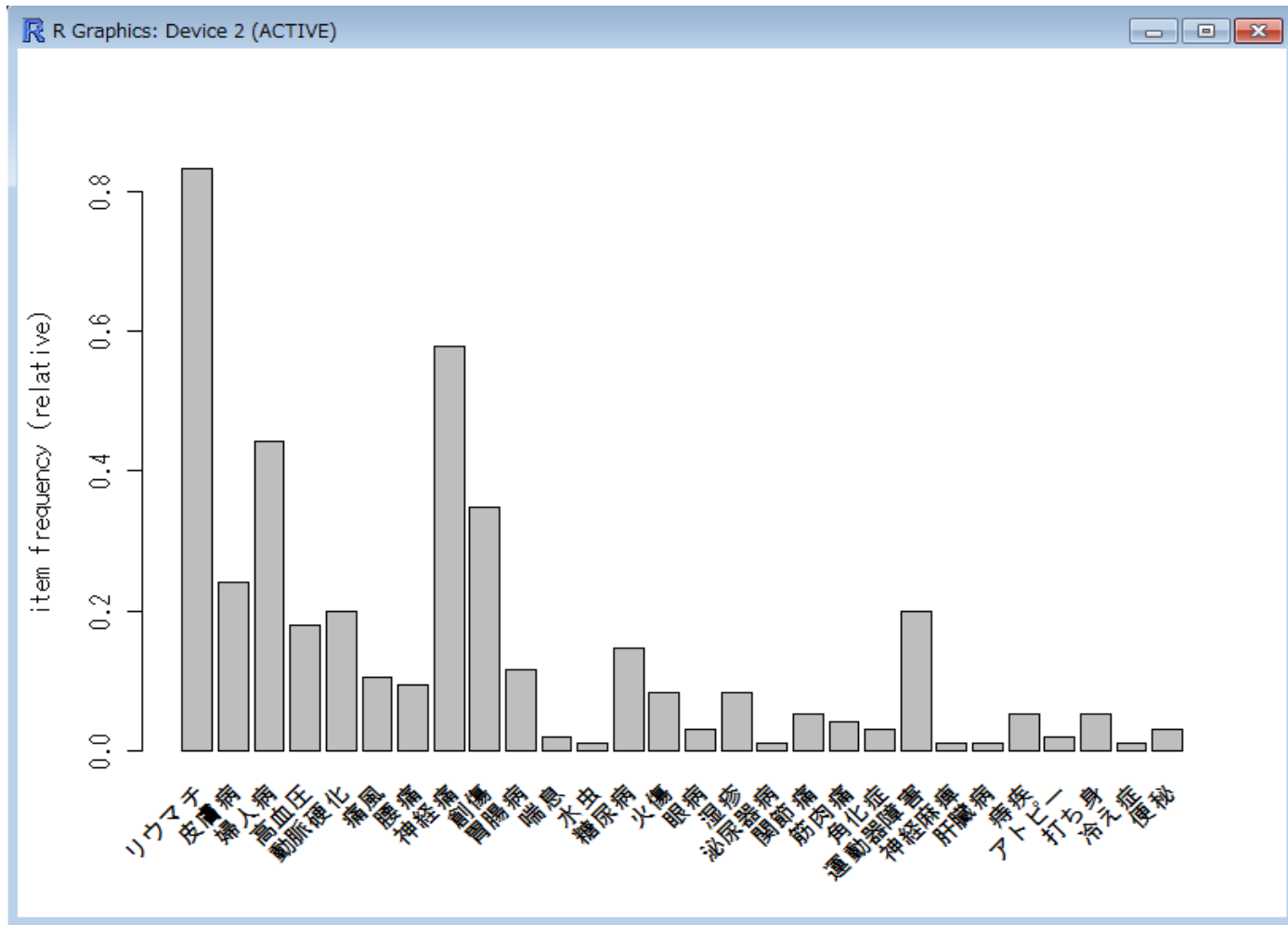
*itemFrequency()

この場合は疾患の出現確率

* itemFrequencyPlot()

棒グラフとして表示

疾患の出現確率の棒グラフ



連関規則の構築と抽出

- > 温泉ルール <- apriori(温泉トランザクション,
parameter=list(maxlen=4,support=0.04,
confidence=0.55, ext=TRUE))

- *apriori() 連関規則を抽出する関数
- *maxlen 一つの連関規則に含まれる最大項目数
- *support 構築する規則のサポートの下限の値
- *confidence 構築する規則の信頼度の下限の値
- *ext=TRUE 前提確率をしめず lhs.supportも結果に表示

構築されたルールの表示

```
R Console
> summary(温泉ルール)
set of 134 rules

rule length distribution (lhs + rhs):sizes
 1  2  3  4
2 34 63 35

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 1.000  2.000  3.000  2.978  4.000  4.000

summary of quality measures:
  support      confidence    lhs.support      lift
Min.   :0.04211   Min.   :0.5625   Min.   :0.04211   Min.   : 0.6872
1st Qu.:0.04211   1st Qu.:0.6667   1st Qu.:0.05263   1st Qu.: 1.2025
Median :0.07368   Median :0.8750   Median :0.09474   Median : 1.3975
Mean   :0.10322   Mean   :0.8397   Mean   :0.13056   Mean   : 2.1557
3rd Qu.:0.10526   3rd Qu.:1.0000   3rd Qu.:0.14737   3rd Qu.: 1.7812
Max.   :0.83158   Max.   :1.0000   Max.   :1.00000   Max.   :19.0000

mining info:
  data ntransactions support confidence
温泉トランザクション      95    0.04    0.55
> |
```



```

R Console
> inspect(温泉ルール)

```

	lhs	rhs	support	confidence	lhs.support	lift
1		[神経痛]	0.57894737	0.5789474	1.0000000	1.0000000
2		[リュマチ]	0.83157895	0.8315789	1.0000000	1.0000000
3	[筋肉痛]	[関節痛]	0.04210526	1.0000000	0.04210526	19.0000000
4	[関節痛]	[筋肉痛]	0.04210526	0.8000000	0.05263158	19.0000000
5	[筋肉痛]	[神経痛]	0.04210526	1.0000000	0.04210526	1.7272727
6	[関節痛]	[神経痛]	0.05263158	1.0000000	0.05263158	1.7272727
7	[痔疾]	[神経痛]	0.04210526	0.8000000	0.05263158	1.3818182
8	[湿疹]	[創傷]	0.05263158	0.6250000	0.08421053	1.7992424
9	[湿疹]	[リュマチ]	0.07368421	0.8750000	0.08421053	1.0522152
10	[腰痛]	[神経痛]	0.08421053	0.8888889	0.09473684	1.5353535
11	[腰痛]	[リュマチ]	0.08421053	0.8888889	0.09473684	1.0689170
12	[火傷]	[皮膚病]	0.05263158	0.6250000	0.08421053	2.5815217
13	[火傷]	[創傷]	0.07368421	0.8750000	0.08421053	2.5189394
14	[火傷]	[リュマチ]	0.05263158	0.6250000	0.08421053	0.7515823
15	[痛風]	[神経痛]	0.06315789	0.6000000	0.10526316	1.0363636
16	[痛風]	[リュマチ]	0.10526316	1.0000000	0.10526316	1.2025316
17	[胃腸病]	[婦人病]	0.09473684	0.8181818	0.11578947	1.8506494
18	[胃腸病]	[神経痛]	0.10526316	0.9090909	0.11578947	1.5702479
19	[胃腸病]	[リュマチ]	0.11578947	1.0000000	0.11578947	1.2025316
20	[高血圧]	[動脈硬化]	0.15789474	0.8823529	0.17894737	4.4117647

```

R Console
> inspect(温泉ルール[c(20,21,70,69,3,4,127,83,61)])

```

	lhs	rhs	support	confidence	lhs.support	lift
1	[高血圧]	[動脈硬化]	0.15789474	0.8823529	0.17894737	4.411765
2	[動脈硬化]	[高血圧]	0.15789474	0.7894737	0.2000000	4.411765
3	[リュマチ, 動脈硬化]	[高血圧]	0.12631579	0.8571429	0.14736842	4.789916
4	[リュマチ, 高血圧]	[動脈硬化]	0.12631579	0.8571429	0.14736842	4.285714
5	[筋肉痛]	[関節痛]	0.04210526	1.0000000	0.04210526	19.000000
6	[関節痛]	[筋肉痛]	0.04210526	0.8000000	0.05263158	19.000000
7	[リュマチ, 婦人病, 創傷]	[運動器障害]	0.09473684	0.6428571	0.14736842	3.214286
8	[婦人病, 創傷]	[運動器障害]	0.09473684	0.5625000	0.16842105	2.812500
9	[神経痛, 胃腸病]	[婦人病]	0.09473684	0.9000000	0.10526316	2.035714

シングルモルトウイスキーの分析

	A	B	C	D	E	F
1	名前	色	香り	口当たり	味	後味
2	アバフェル	黄色	フルーツ	中程度	オイリィ	香辛料
3	アベラワー	赤色	豊潤	スムーズ	香辛料	あっさり
4	アードベック	シェリー	塩っばい	しっかり	甘い	塩っばい
5	アードモア	青金色	シェリー	充実した	甘い	大きな
6	オーヘント	青金色	芝生	ソフト	甘い	香辛料
7	オルトモー	薄青白色	新鮮	しっかり	香辛料	乾いた
8	バルブレア	白ワイン	フルーツ	軽やか	香辛料	あっさり
9	バルミニック	青金色	芝生	軽やか	甘い	余韻ある
10	バルヴィニ	琥珀色	フルーツ	中程度	甘い	あっさり
11	パンフ	濃金色	ビート	中程度	甘い	香辛料
12	ベン・ネヴィ	濃琥珀色	芝生	オイリィ	甘い	甘い
13	ベンリアック	金色	芝生	軽やか	甘い	スモーク
14	ベンリネス	琥珀色	ビート	しっかり	乾いた	スモーク
15	ベンローマ	琥珀色	芝生	中程度	甘い	スモーク
16	ブランドノッ	黄色	芝生	しっかり	充実した	フルーツ
17	ブレア・アン	青金色	香辛料	軽やか	アロマ	香辛料
18	ボウモア	古金色	塩っばい	軽やか	甘い	名残惜く
19	ブラックラ	古金色	甘い	充実した	アロマ	芝生
20	ブルイック	青白色	軽やか	しっかり	甘い	充実した
21	ブナハーブ	金色	塩っばい	しっかり	甘い	充実した
22	カリラ	白ワイン	塩っばい	オイリィ	香辛料	香辛料
23	キャバドニ	濃金色	ビート	軽やか	甘い	あっさり
24	カードウ	青白色	軽やか	軽やか	甘い	あっさり

```
R Console
> library(arules)
> スコッチデータ <- read.csv("scotch.csv",header=TRUE,row.names=1)
> スコッチトランザクション <- as(スコッチデータ,"transactions")
> summary(スコッチトランザクション)
transactions as itemMatrix in sparse format with
 109 rows (elements/itemsets/transactions) and
 64 columns (items) and a density of 0.078125

most frequent items:
      味=甘い  □当たり=軽やか      色=濃金色  後味=余韻ある  香り=シェリー
           51          34          27          24          23
      (Other)
       386

element (itemset/transaction) length distribution:
sizes
 5
109

      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
       5         5         5         5         5         5

includes extended item information - examples:
      labels variables  levels
1 色=シェリー      色 シェリー
2 色=黄色      色 黄色
3 色=金色      色 金色

includes extended transaction information - examples:
      transactionID
1 アバフェルディ
2 アベラワー
3 アードベッグ
> |
```

* 文字列のデータはそのままトランザクションデータに変更

```

R Console
> itemFrequency(スコッチランザクション)
色=シェリー          色=黄色          色=金色          色=古金色
0.009174312          0.018348624      0.192660550      0.018348624
色=青金色          色=青銅色        色=青白色        色=青白琥珀
0.091743119          0.045871560      0.036697248      0.036697248
色=赤色            色=濃琥珀色      色=濃琥珀色      色=白ワイン
0.036697248          0.247706422      0.036697248      0.073394495
色=薄青白色        色=琥珀色        香り=シェリー    香り=ドライ
0.018348624          0.137614679      0.211009174      0.073394495
香り=ビート        香り=フルーツ    香り=塩っ(まい) 香り=甘い
0.045871560          0.100917431      0.110091743      0.064220183
香り=軽やか        香り=香辛料      香り=芝生        香り=新鮮
0.036697248          0.045871560      0.146788991      0.009174312
香り=豊潤          □当たり=オイリィ □当たり=しっかり □当たり=スモーズ
0.155963303          0.165137615      0.211009174      0.119266055
□当たり=ソフト      □当たり=軽やか   □当たり=充実した □当たり=熟成した
0.009174312          0.311926606      0.100917431      0.009174312
□当たり=中程度      味=アロマ        味=オイリィ      味=シェリー
0.073394495          0.100917431      0.055045872      0.009174312
味=すっきり        味=スムーズ      味=スモーク      味=塩っ(まい)
0.009174312          0.009174312      0.027522936      0.110091743
味=乾いた          味=甘い          味=香辛料        味=芝生
0.018348624          0.467889908      0.137614679      0.045871560
味=充実した        後味=あっさり    後味=アロマ      後味=オイリィ
0.009174312          0.100917431      0.045871560      0.018348624
後味=すっきり      後味=スムーズ    後味=スモーク    後味=フルーツ
0.009174312          0.018348624      0.073394495      0.045871560
後味=ほてった      後味=塩っ(まい) 後味=乾いた      後味=甘い
0.018348624          0.027522936      0.055045872      0.045871560
後味=軽やか        後味=香辛料      後味=芝生        後味=充実した
0.018348624          0.100917431      0.055045872      0.027522936
後味=大きな        後味=長い余韻    後味=名残惜く    後味=余韻ある
0.027522936          0.036697248      0.055045872      0.220183486
> |

```

連関規則の条件を設定

- > スコッチルール <- apriori(スコッチトランザクション,
parameter=list(maxlen=3, support=0.02,
confidence=0.35, ext=TRUE))
- > 甘いかつ改善率1より大ルール <- subset(
スコッチルール,subset=(rhs %in% "味=甘い")&(lift>1.0))

* subset() 一度構築された連関規則から、
さらに条件を設定してルールを抽出する関数

新たに抽出したルール

```
R Console
> 甘いかつ改善率1より大ルール <- subset(スコッチルール,subset=(rhs %in% "味=甘い")&(lift>1.0))
> inspect(甘いかつ改善率1より大ルール)
  lhs                                     rhs      support confidence lhs.support lift
1  [色=青白色] => [味=甘い] 0.02752294 0.7500000 0.03669725 1.602941
2  [香り=ピート] => [味=甘い] 0.03669725 0.8000000 0.04587156 1.709804
3  [後味=甘い] => [味=甘い] 0.03669725 0.8000000 0.04587156 1.709804
4  [色=青銅色] => [味=甘い] 0.02752294 0.6000000 0.04587156 1.282353
5  [後味=名残惜く] => [味=甘い] 0.03669725 0.6666667 0.05504587 1.424837
6  [後味=芝生] => [味=甘い] 0.02752294 0.5000000 0.05504587 1.068627
7  [口当たり=中程度] => [味=甘い] 0.03669725 0.5000000 0.07339450 1.068627
8  [香り=ドライ] => [味=甘い] 0.03669725 0.5000000 0.07339450 1.068627
9  [後味=スモーク] => [味=甘い] 0.03669725 0.5000000 0.07339450 1.068627
10 [後味=香辛料] => [味=甘い] 0.05504587 0.5454545 0.10091743 1.165775
11 [香り=芝生] => [味=甘い] 0.08256881 0.5625000 0.14678899 1.202206
12 [香り=豊潤] => [味=甘い] 0.08256881 0.5294118 0.15596330 1.131488
13 [色=金色] => [味=甘い] 0.09174312 0.4761905 0.19266055 1.017740
14 [口当たり=しっかり] => [味=甘い] 0.10091743 0.4782609 0.21100917 1.022165
15 [色=濃金色] => [味=甘い] 0.13761468 0.5555556 0.24770642 1.187364
16 [口当たり=軽やか] => [味=甘い] 0.17431193 0.5588235 0.31192661 1.194348
17 [色=濃金色,
  香り=ピート] => [味=甘い] 0.03669725 1.0000000 0.03669725 2.137255
18 [香り=ドライ,
  口当たり=軽やか] => [味=甘い] 0.02752294 0.7500000 0.03669725 1.602941
19 [色=琥珀色,
  後味=あっさり] => [味=甘い] 0.02752294 0.7500000 0.03669725 1.602941
20 [口当たり=軽やか,
  後味=あっさり] => [味=甘い] 0.02752294 0.6000000 0.04587156 1.282353
21 [香り=豊潤,
```

天気の分析

	A	B	C	D	E	F	G
1	年月日	月	当日天気	前日天気	前々日天気	三日前天気	一週間前天気
2	961231	12	晴	晴	晴	晴	晴
3	961230	12	晴	晴	晴	晴	晴
4	961229	12	晴	晴	晴	晴	晴
5	961228	12	晴	晴	晴	晴	晴
6	961227	12	晴	晴	晴	晴	晴
7	961226	12	晴	晴	晴	晴	晴
8	961225	12	晴	晴	晴	晴	雨
9	961224	12	晴	晴	晴	晴	雨
10	961223	12	晴	晴	晴	晴	晴
11	961222	12	晴	晴	晴	晴	晴
12	961221	12	晴	晴	晴	雨	晴
13	961220	12	晴	晴	雨	雨	曇
14	961219	12	晴	雨	雨	晴	晴
15	961218	12	雨	雨	晴	晴	雨
16	961217	12	雨	晴	晴	晴	晴
17	961216	12	晴	晴	晴	曇	晴
18	961215	12	晴	晴	曇	晴	晴
19	961214	12	晴	曇	晴	雨	晴
20	961213	12	曇	晴	雨	晴	晴
21	961212	12	晴	雨	晴	晴	雨
22	961211	12	雨	晴	晴	晴	晴
23	961210	12	晴	晴	晴	晴	晴
24	961209	12	晴	晴	晴	晴	晴
25	961208	12	晴	晴	晴	雨	晴
26	961207	12	晴	晴	雨	晴	晴
27	961206	12	晴	雨	晴	晴	晴
28	961205	12	雨	晴	晴	晴	晴
29	961204	12	晴	晴	晴	晴	雨
30	961203	12	晴	晴	晴	晴	晴

```
R Console
> library(arules)
> 天気データ <- read.csv("tenki.csv",header=TRUE,row.names=1,colClasses="factor")
> 天気トランザクション <- as(天気データ,"transactions")
> summary(天気トランザクション)
transactions as itemMatrix in sparse format with
3643 rows (elements/itemsets/transactions) and
32 columns (items) and a density of 0.1875

most frequent items:
  当日天気=晴   前日天気=晴  前々日天気=晴  三日前天気=晴  一週間前天気=晴   (Other)
      1962         1962         1961         1960         1958         12055

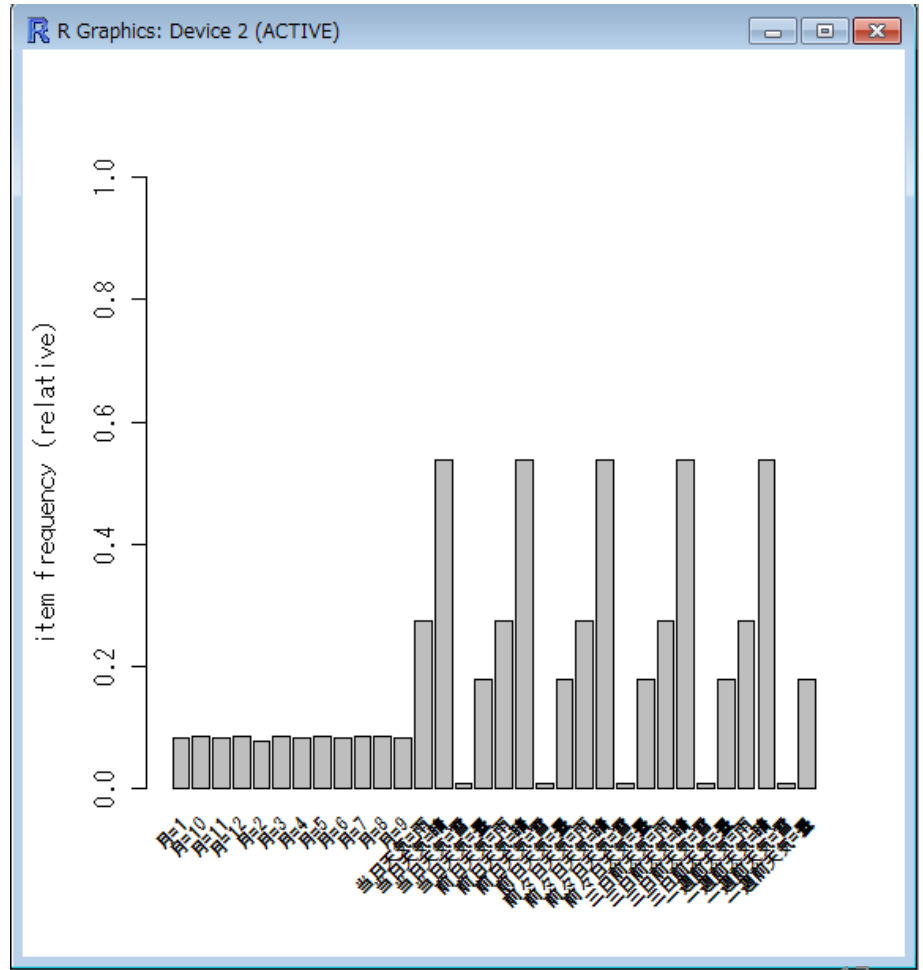
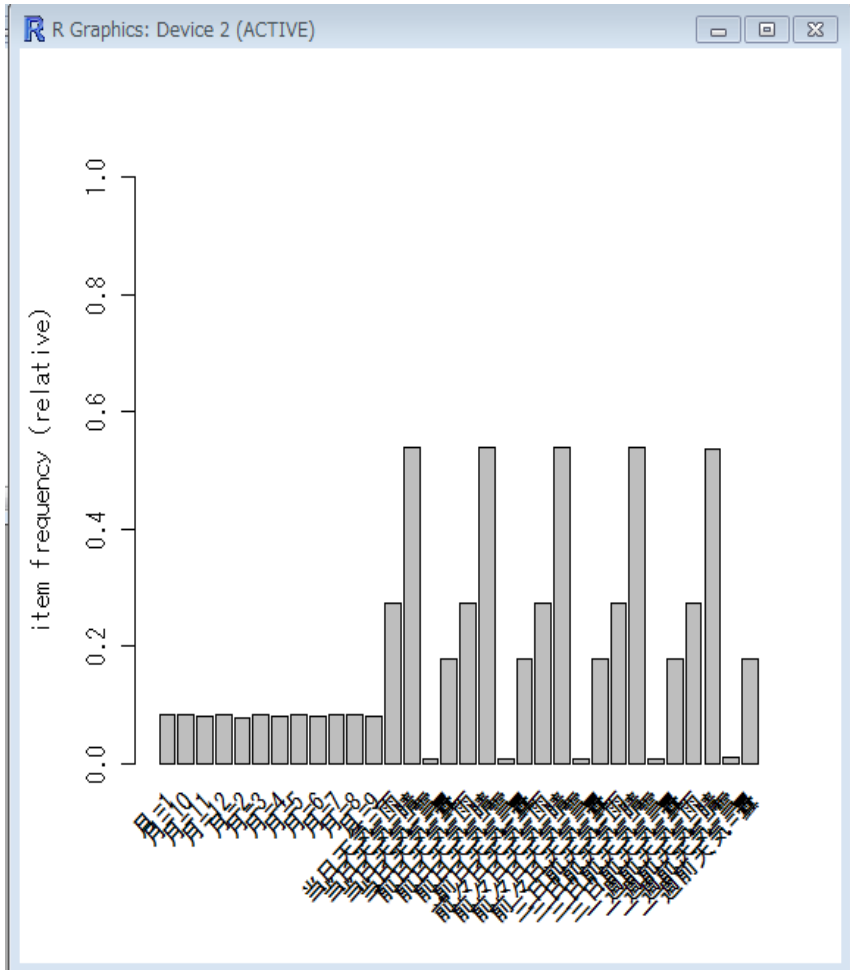
element (itemset/transaction) length distribution:
sizes
  6
3643

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    6         6         6         6         6         6

includes extended item information - examples:
  labels variables levels
1   月=1         月         1
2   月=10        月         10
```

*colClasses="factor" データに数字と文字が混在しているときに読み込む命令


```
R Console
> itemFrequencyPlot(天気トランザクション,ylim=c(0,1))
> itemFrequencyPlot(天気トランザクション,ylim=c(0,1),cex.names=0.7)
> |
```



ルール構築時に条件を設定(1)

- > 晴れルール <- apriori(天気ランザクション,
parameter=list(maxlen=5, support=0.05, confidence=0.65, ext=TRUE),
appearance=list(rhs="当日天気=晴", default="lhs"))
- *appearance=list() ルールを構築する時点でルール・ヘッドやルール・ボディ
におかれる項目を指定するもの。
- > inspect(SORT(晴れルール,"support")[1:10,])
- *inspect中のSORT() 構築されたルールをサポート、信頼度、リフト等
によって並べ替える関数

ルール構築時に条件を設定(2)

```
R Console
> inspect(SORT(晴れルール,"support")[1:10, ])
  lhs                rhs                support confidence lhs.support    lift
1  [前日天気=晴]    => [当日天気=晴] 0.36782871 0.6829766 0.5385671 1.268136
2  [前日天気=晴,   => [当日天気=晴] 0.26132309 0.7109783 0.3675542 1.320130
   前々日天気=晴]
3  [前日天気=晴,   => [当日天気=晴] 0.23552018 0.7161937 0.3288498 1.329813
   三日前天気=晴]
4  [前日天気=晴,   => [当日天気=晴] 0.22261872 0.7058312 0.3153994 1.310572
   一週前天気=晴]
5  [前日天気=晴,   => [当日天気=晴] 0.19297282 0.7392219 0.2610486 1.372571
   前々日天気=晴,
   三日前天気=晴]
6  [前日天気=晴,   => [当日天気=晴] 0.16277793 0.7330037 0.2220697 1.361026
   前々日天気=晴,
   一週前天気=晴]
7  [前日天気=晴,   => [当日天気=晴] 0.15371946 0.7417219 0.2072468 1.377213
   三日前天気=晴,
   一週前天気=晴]
8  [前々日天気=晴, => [当日天気=晴] 0.15289596 0.6910670 0.2212462 1.283159
   三日前天気=晴,
   一週前天気=晴]
9  [前日天気=晴,   => [当日天気=晴] 0.12709306 0.7615132 0.1668954 1.413961
   前々日天気=晴,
   三日前天気=晴,
   一週前天気=晴]
10 [月=12]          => [当日天気=晴] 0.06807576 0.8000000 0.0850947 1.485423
> |
```

条件設定の応用(1)

- > 雪ルール <- apriori(天気トランザクション,
parameter=list(maxlen=3, support=0.0001, confidence=0.01, ext=TRUE),
appearance=list(rhs="当日天気=雪", default="lhs"))

> 雪ルール1.2月 <- subset(雪ルール,
subset=(lhs %in% "月=1") |(lhs %in% "月=2"))

条件設定の応用(2)

```
R Console
> 雪ルール1.2月 <- subset(雪ルール, subset= (lhs %in% "月=1") |(lhs %in% "月=2"))
> inspect(雪ルール1.2月[1:10, ])
  lhs                rhs                support confidence lhs.support    lift
1 {月=2}            => {当日天気=雪} 0.0038429865 0.05000000 0.076859731  5.519697
2 {月=1}            => {当日天気=雪} 0.0027449904 0.03300330 0.083173209  3.643364
3 {月=2,
  前日天気=雪}    => {当日天気=雪} 0.0013724952 0.31250000 0.004391985 34.498106
4 {月=1,
  前日天気=雪}    => {当日天気=雪} 0.0002744990 0.12500000 0.002195992 13.799242
5 {月=2,
  前々日天気=雪} => {当日天気=雪} 0.0005489981 0.13333333 0.004117486 14.719192
6 {月=1,
  三日前天気=雪} => {当日天気=雪} 0.0002744990 0.11111111 0.002470491 12.265993
7 {月=1,
  一週前天気=雪} => {当日天気=雪} 0.0005489981 0.18181818 0.003019489 20.071625
8 {月=2,
  前日天気=曇}    => {当日天気=雪} 0.0008234971 0.13043478 0.006313478 14.399209
9 {月=2,
  前々日天気=曇} => {当日天気=雪} 0.0008234971 0.13043478 0.006313478 14.399209
10 {月=2,
    三日前天気=曇} => {当日天気=雪} 0.0005489981 0.08695652 0.006313478  9.599473
> |
```

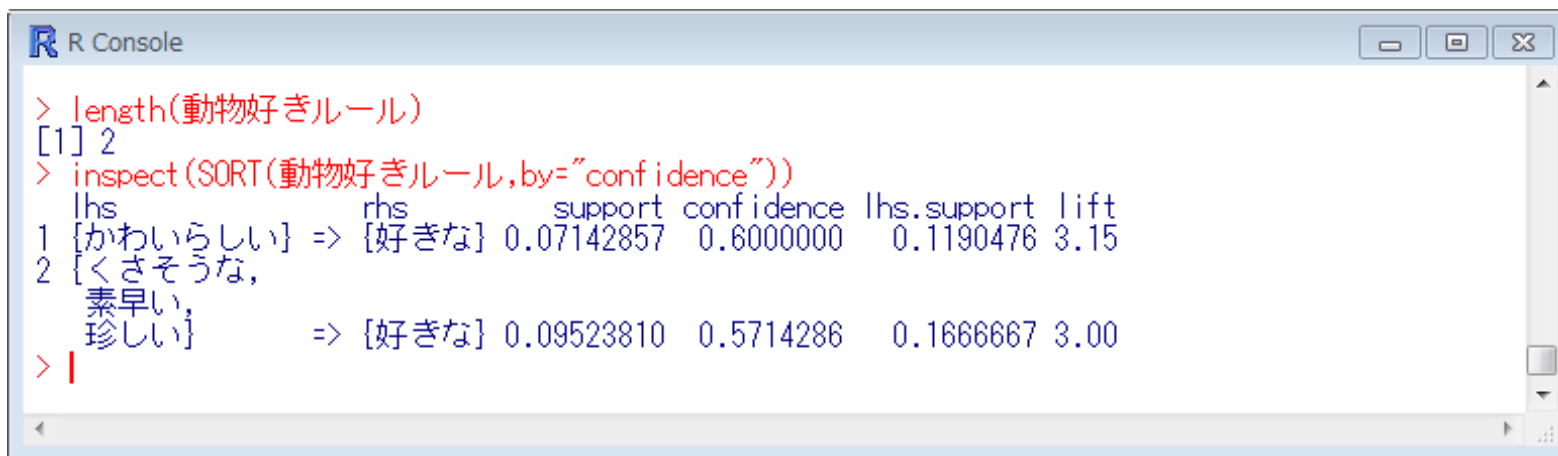
動物に対する印象データの分析(1)

	A	B	C	D	E	F	G	H	I	J
1		賢い	くさそうな	好きな	かわいらしい	大きな	危険な	素早い	美味しそう	珍しい
2	ライオン	0	1	0	0	1	1	1	0	1
3	ぞう	0	1	0	0	1	1	0	0	1
4	うし	0	1	0	0	0	0	0	1	0
5	しまうま	0	1	0	0	0	0	1	0	1
6	ヒト	1	0	0	0	0	0	0	0	0
7	チーター	0	1	1	0	0	1	1	0	1
8	カピバラ	0	1	0	0	0	0	0	0	1
9	ねずみ	1	1	0	0	0	0	1	0	0
10	すずめ	0	0	0	1	0	0	0	0	0
11	からす	1	1	0	0	0	0	0	0	0
12	りす	0	0	1	1	0	0	0	0	0
13	さい	0	1	0	0	1	1	0	0	1
14	ごりら	1	1	0	0	0	1	1	0	1
15	くじら	0	0	1	0	1	1	0	0	1
16	わし	0	1	1	0	0	0	1	0	1
17	むささび	0	1	1	1	0	0	1	0	1
18	アイアイ	0	0	0	0	0	0	1	0	1
19	にわとり	0	1	0	0	0	0	0	1	0
20	ぶた	0	1	0	0	0	0	0	1	0
21	カエル	0	1	0	0	0	0	0	1	0
22	カンガルー	0	1	1	0	0	0	0	0	1
23	オランウー	1	1	0	0	0	0	0	0	1
24	モルモット	0	0	0	0	0	0	0	0	0
25	オオカミ	0	1	1	0	0	1	1	0	1
26	ペンギン	0	0	1	1	0	0	0	0	1
27	かば	0	1	0	0	1	0	0	0	1
28	いのしし	0	1	0	0	0	0	0	1	1
29	しか	0	0	0	0	0	0	0	1	0
30	フニ	0	0	0	0	0	0	0	1	1

動物に対する印象データの分析(2)

- > 動物好きルール <- apriori(動物トランザクション,
parameter=list(maxlen=4, support=0.05, confidence=0.55, ext=TRUE),
appearance=list(rhs="好きな", default="lhs"))
> length(動物好きルール)
> inspect(SORT(動物好きルール, by="confidence"))

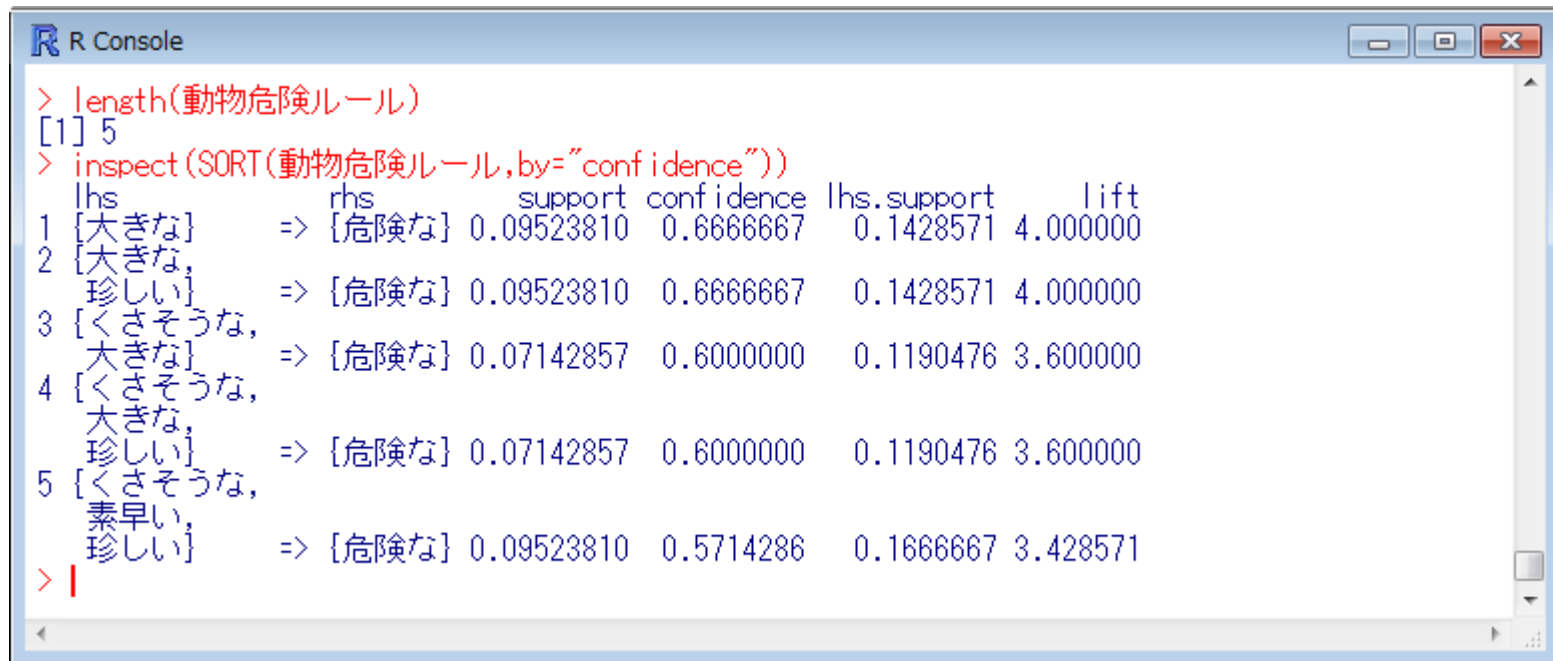
*length() 構築されたルールの数を出力。



```
R Console
> length(動物好きルール)
[1] 2
> inspect(SORT(動物好きルール, by="confidence"))
  lhs                rhs      support confidence lhs.support lift
1 [かわいらしい] => [好きな] 0.07142857  0.6000000  0.1190476 3.15
2 [くさそうな,
   素早い,
   珍しい]         => [好きな] 0.09523810  0.5714286  0.1666667 3.00
> |
```

動物に対する印象データの分析(3)

- > 動物危険ルール <- apriori(動物トランザクション,
parameter=list(maxlen=4, support=0.05, confidence=0.55, ext=TRUE),
appearance=list(rhs="危険な", default="lhs"))



```
> length(動物危険ルール)
[1] 5
> inspect(SORT(動物危険ルール, by="confidence"))
  lhs                rhs      support confidence lhs.support  lift
1 {大きな}          => {危険な} 0.09523810  0.6666667  0.1428571  4.000000
2 {大きな,          => {危険な} 0.09523810  0.6666667  0.1428571  4.000000
  珍しい}
3 {くさそうな,     => {危険な} 0.07142857  0.6000000  0.1190476  3.600000
  大きな}
4 {くさそうな,     => {危険な} 0.07142857  0.6000000  0.1190476  3.600000
  大きな,
  珍しい}
5 {くさそうな,     => {危険な} 0.09523810  0.5714286  0.1666667  3.428571
  素早い,
  珍しい}
> |
```