Association Rules Example

Apriori

Preparing the dataset:

set working directory and other setup:

```
#setwd("/Users/jacobmontielbravo/Desktop/portfolio-example")
setwd("C:/Users/Jacob Montiel-Bravo/Box/Work")
#install.packages('arules')
```

Creating data set

```
pokemongames <- read.csv('example-data.csv', header = TRUE)
pokemongames <- pokemongames[-c(1)]
View(pokemongames)


write.csv(pokemongames, "pokemongames-new.csv")
```

We need to convert the dataset into a "Sparse Matrix"

```
library(arules)
```

```
## Warning: package 'arules' was built under R version 4.0.5

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.0.5

##
## Attaching package: 'arules'

## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

```
pokemongames = read.transactions('pokemongames-new.csv', sep = ',', rm.duplicates = TRUE)
```

note that you see the message in the console: "distribution of transaction with duplicates' this indicates that there were xxx transactions with 1 duplicate A duplicate would simply indicate that someone used the same term twice in a post.... ...which is probably not very significant

Viewing data

```r
summary(pokemongames)
```

```
## transactions as itemMatrix in sparse format with
##  1802 rows (elements/itemsets/transactions) and
##  1819 columns (items) and a density of 0.001105912
##
## most frequent items:
## Scarlet   White   Sword  Violet    Moon (Other)
##     371     324     243     224     173    2290
##
## element (itemset/transaction) length distribution:
## sizes
##   1   2   3   4   5   6   7   8   9  14  15
## 670 715 252 102  44  10   5   1   1   1   1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   2.000   2.012   2.000  15.000
##
## includes extended item information - examples:
##   labels
## 1      1
## 2     10
## 3    100
```

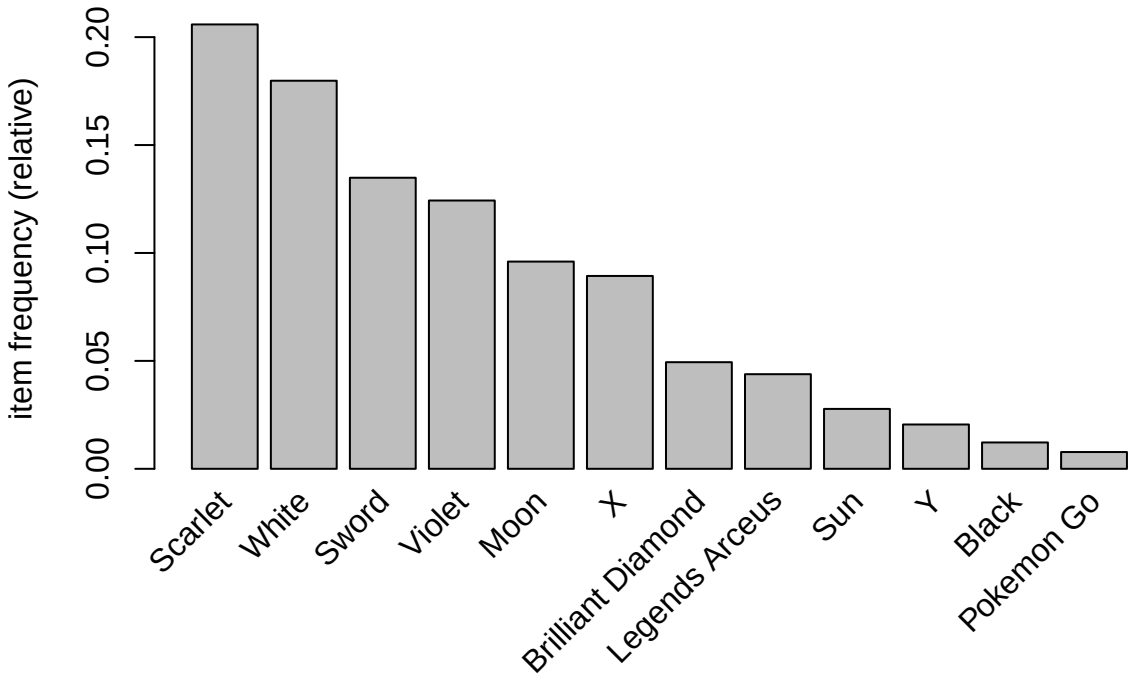
```r
head(pokemongames)
```

```
## transactions in sparse format with
##  6 transactions (rows) and
##  1819 items (columns)
```

```r
#View(pokemongames)
```

Note that "density" (see console) indicates the percentage of non-zero values

```r
itemFrequencyPlot(pokemongames, topN = 12) #topN is how many of the top purchased products you want (we
```



Definitions

```
support gives us the percent of occurrences that the items appear--we may...
...not want to look at terms that haven't appeared very much as a total of all terms
Confidence gives us the number of times that terms x and y appeared together...
  ...divided by the number of times y appeared
Coverage (also called cover or LHS-support) is the support of the left-hand-side of the rule X => Y, i.
Count is simply the total number of records where this rule has been observed
the maxlen= parameter in the algorithm specifies the maximum number of items on the lhs
verbose shows the number of iterations
```

Training Apriori on the dataset

```
#non-specific approach:
rules = apriori(data = pokemongames, parameter = list(support = 0.003, confidence = 0.4))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5   0.003      1
##  maxlen target  ext
##      10  rules TRUE
```

```
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 5
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[1819 item(s), 1802 transaction(s)] done [0.00s].
## sorting and recoding items ... [14 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [46 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

note that we get a lot more rules, but we only want to look at the 50 highest:

```
?apriori()
```

```
## starting httpd help server ... done
```

```
inspect(sort(rules, by = 'count')[1:20])
```

```
##      lhs                              rhs              support     confidence
## [1]  {Brilliant Diamond}           => {White}          0.031631521 0.6404494
## [2]  {Y}                           => {White}          0.011098779 0.5405405
## [3]  {Y}                           => {Scarlet}        0.008324084 0.4054054
## [4]  {Brilliant Diamond, Scarlet} => {White}          0.008324084 0.6250000
## [5]  {Scarlet, X}                  => {White}          0.008324084 0.4545455
## [6]  {Black}                       => {Y}              0.007769145 0.6363636
## [7]  {Brilliant Diamond, Violet}  => {White}          0.007214206 0.5909091
## [8]  {Black}                       => {White}          0.006104329 0.5000000
## [9]  {Moon, Sword}                 => {Scarlet}        0.006104329 0.4074074
## [10] {Black}                       => {Legends Arceus} 0.005549390 0.4545455
## [11] {Black}                       => {X}              0.004994451 0.4090909
## [12] {Black}                       => {Scarlet}        0.004994451 0.4090909
## [13] {Black, Y}                    => {Legends Arceus} 0.004994451 0.6428571
## [14] {Black, Legends Arceus}       => {Y}              0.004994451 0.9000000
## [15] {Legends Arceus, Y}           => {Black}          0.004994451 0.6923077
## [16] {Black, Y}                    => {Scarlet}        0.004994451 0.6428571
## [17] {Black, Scarlet}              => {Y}              0.004994451 1.0000000
## [18] {Scarlet, Y}                  => {Black}          0.004994451 0.6000000
## [19] {Scarlet, Y}                  => {White}          0.004994451 0.6000000
## [20] {White, Y}                    => {Scarlet}        0.004994451 0.4500000
##      coverage    lift      count
## [1]  0.049389567  3.562006 57
## [2]  0.020532741  3.006340 20
## [3]  0.020532741  1.969112 15
## [4]  0.013318535  3.476080 15
## [5]  0.018312986  2.528058 15
## [6]  0.012208657 30.992629 14
## [7]  0.012208657  3.286476 13
```

```
## [8]  0.012208657  2.780864 11
## [9]  0.014983352  1.978836 11
## [10] 0.012208657 10.368239 10
## [11] 0.012208657  4.578769  9
## [12] 0.012208657  1.987013  9
## [13] 0.007769145 14.663653  9
## [14] 0.005549390 43.832432  9
## [15] 0.007214206 56.706294  9
## [16] 0.007769145  3.122449  9
## [17] 0.004994451 48.702703  9
## [18] 0.008324084 49.145455  9
## [19] 0.008324084  3.337037  9
## [20] 0.011098779  2.185714  9
```

What if we consider items that are mentioned with more frequency? We raise "support"

```
rules = apriori(data = pokemongames, parameter = list(support = 0.004, confidence = 0.2))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5   0.004      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 7
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[1819 item(s), 1802 transaction(s)] done [0.00s].
## sorting and recoding items ... [14 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [76 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
inspect(sort(rules, by = 'lift')[1:20])
```

```
##      lhs                           rhs                 support     confidence
## [1]  {Legends Arceus, Y}        => {Black}             0.004994451 0.6923077
## [2]  {Scarlet, Y}               => {Black}             0.004994451 0.6000000
## [3]  {Black, Scarlet}           => {Y}                 0.004994451 1.0000000
## [4]  {Black, Legends Arceus}    => {Y}                 0.004994451 0.9000000
## [5]  {Y}                        => {Black}             0.007769145 0.3783784
## [6]  {Black}                    => {Y}                 0.007769145 0.6363636
## [7]  {Black, Y}                 => {Legends Arceus}    0.004994451 0.6428571
## [8]  {Black}                    => {Legends Arceus}    0.005549390 0.4545455
## [9]  {Y}                        => {Legends Arceus}    0.007214206 0.3513514
```

```
## [10] {Scarlet, White}             => {Brilliant Diamond} 0.008324084 0.3125000
## [11] {Violet, White}              => {Brilliant Diamond} 0.007214206 0.2280702
## [12] {Black}                      => {X}                 0.004994451 0.4090909
## [13] {Brilliant Diamond, X}       => {White}             0.004439512 0.7272727
## [14] {Brilliant Diamond}          => {White}             0.031631521 0.6404494
## [15] {Violet, White}              => {X}                 0.009988901 0.3157895
## [16] {Scarlet, White}             => {X}                 0.008324084 0.3125000
## [17] {Brilliant Diamond, Scarlet} => {White}             0.008324084 0.6250000
## [18] {Scarlet, Y}                 => {White}             0.004994451 0.6000000
## [19] {Y}                          => {X}                 0.006104329 0.2972973
## [20] {Brilliant Diamond, Violet}  => {White}             0.007214206 0.5909091
##      coverage    lift      count
## [1]  0.007214206 56.706294  9
## [2]  0.008324084 49.145455  9
## [3]  0.004994451 48.702703  9
## [4]  0.005549390 43.832432  9
## [5]  0.020532741 30.992629 14
## [6]  0.012208657 30.992629 14
## [7]  0.007769145 14.663653  9
## [8]  0.012208657 10.368239 10
## [9]  0.020532741  8.014369 13
## [10] 0.026637070  6.327247 15
## [11] 0.031631521  4.617780 13
## [12] 0.012208657  4.578769  9
## [13] 0.006104329  4.044893  8
## [14] 0.049389567  3.562006 57
## [15] 0.031631521  3.534488 18
## [16] 0.026637070  3.497671 15
## [17] 0.013318535  3.476080 15
## [18] 0.008324084  3.337037  9
## [19] 0.020532741  3.327514 11
## [20] 0.012208657  3.286476 13
```

'Scarlet'

```
rules <- apriori(data=pokemongames, parameter=list (support=0.0001, confidence=0.01, maxlen=2), appearar
inspect(sort(rules, by = 'count')[1:20])
```

```
##      lhs                   rhs       support     confidence coverage
## [1]  {}                 => {Scarlet} 0.205882353 0.2058824  1.000000000
## [2]  {Violet}           => {Scarlet} 0.026637070 0.2142857  0.124306326
## [3]  {White}            => {Scarlet} 0.026637070 0.1481481  0.179800222
## [4]  {Sword}            => {Scarlet} 0.019422863 0.1440329  0.134850166
## [5]  {X}                => {Scarlet} 0.018312986 0.2049689  0.089345172
## [6]  {Moon}             => {Scarlet} 0.016093230 0.1676301  0.096004440
## [7]  {Brilliant Diamond} => {Scarlet} 0.013318535 0.2696629  0.049389567
## [8]  {Y}                => {Scarlet} 0.008324084 0.4054054  0.020532741
## [9]  {Legends Arceus}   => {Scarlet} 0.007769145 0.1772152  0.043840178
## [10] {Sun}              => {Scarlet} 0.005549390 0.2000000  0.027746948
## [11] {Black}            => {Scarlet} 0.004994451 0.4090909  0.012208657
## [12] {Pokemon Go}       => {Scarlet} 0.002774695 0.3571429  0.007769145
```

```
## [13] {Shield}           => {Scarlet} 0.001664817 0.3750000  0.004439512
## [14] {1563}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [15] {1527}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [16] {1564}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [17] {1533}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [18] {1019}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [19] {1528}             => {Scarlet} 0.000554939 1.0000000  0.000554939
## [20] {1300}             => {Scarlet} 0.000554939 1.0000000  0.000554939
##      lift      count
## [1]  1.0000000 371
## [2]  1.0408163  48
## [3]  0.7195767  48
## [4]  0.6995885  35
## [5]  0.9955634  33
## [6]  0.8142031  29
## [7]  1.3097913  24
## [8]  1.9691120  15
## [9]  0.8607595  14
## [10] 0.9714286  10
## [11] 1.9870130   9
## [12] 1.7346939   5
## [13] 1.8214286   3
## [14] 4.8571429   1
## [15] 4.8571429   1
## [16] 4.8571429   1
## [17] 4.8571429   1
## [18] 4.8571429   1
## [19] 4.8571429   1
## [20] 4.8571429   1
```

'White'

```
rules <- apriori(data=pokemongames, parameter=list (support=0.0001, confidence=0.01, maxlen=3), appearar
inspect(sort(rules, by = 'count')[1:20])
```

```
##      lhs                             rhs     support     confidence coverage
## [1]  {}                           => {White} 0.179800222 0.1798002  1.000000000
## [2]  {Brilliant Diamond}          => {White} 0.031631521 0.6404494  0.049389567
## [3]  {Violet}                     => {White} 0.031631521 0.2544643  0.124306326
## [4]  {Sword}                      => {White} 0.028856826 0.2139918  0.134850166
## [5]  {Scarlet}                    => {White} 0.026637070 0.1293801  0.205882353
## [6]  {X}                          => {White} 0.025527192 0.2857143  0.089345172
## [7]  {Moon}                       => {White} 0.018312986 0.1907514  0.096004440
## [8]  {Legends Arceus}             => {White} 0.016648169 0.3797468  0.043840178
## [9]  {Y}                          => {White} 0.011098779 0.5405405  0.020532741
## [10] {Violet, X}                  => {White} 0.009988901 0.3829787  0.026082131
## [11] {Scarlet, Violet}            => {White} 0.008879023 0.3333333  0.026637070
## [12] {Brilliant Diamond, Scarlet} => {White} 0.008324084 0.6250000  0.013318535
## [13] {Scarlet, X}                 => {White} 0.008324084 0.4545455  0.018312986
## [14] {Sun}                        => {White} 0.007769145 0.2800000  0.027746948
## [15] {Brilliant Diamond, Violet}  => {White} 0.007214206 0.5909091  0.012208657
```

```
## [16] {Black}                    => {White} 0.006104329 0.5000000  0.012208657
## [17] {Sword, Violet}            => {White} 0.005549390 0.2777778  0.019977802
## [18] {Scarlet, Sword}           => {White} 0.005549390 0.2857143  0.019422863
## [19] {Scarlet, Y}               => {White} 0.004994451 0.6000000  0.008324084
## [20] {Moon, Scarlet}            => {White} 0.004994451 0.3103448  0.016093230
##      lift      count
## [1]  1.0000000 324
## [2]  3.5620058  57
## [3]  1.4152612  57
## [4]  1.1901641  52
## [5]  0.7195767  48
## [6]  1.5890653  46
## [7]  1.0609077  33
## [8]  2.1120488  30
## [9]  3.0063397  20
## [10] 2.1300236  18
## [11] 1.8539095  16
## [12] 3.4760802  15
## [13] 2.5280584  15
## [14] 1.5572840  14
## [15] 3.2864759  13
## [16] 2.7808642  11
## [17] 1.5449246  10
## [18] 1.5890653  10
## [19] 3.3370370   9
## [20] 1.7260536   9
```

'Sword'

```
rules <- apriori(data=pokemongames, parameter=list (support=0.0001, confidence=0.1, maxlen=2), appearan
inspect(sort(rules, by = 'count')[1:20])
```

```
##      lhs                    rhs     support     confidence coverage
## [1]  {}                  => {Sword} 0.134850166 0.1348502  1.000000000
## [2]  {White}             => {Sword} 0.028856826 0.1604938  0.179800222
## [3]  {Violet}            => {Sword} 0.019977802 0.1607143  0.124306326
## [4]  {Moon}              => {Sword} 0.014983352 0.1560694  0.096004440
## [5]  {X}                 => {Sword} 0.013873474 0.1552795  0.089345172
## [6]  {Legends Arceus}    => {Sword} 0.008324084 0.1898734  0.043840178
## [7]  {Brilliant Diamond} => {Sword} 0.005549390 0.1123596  0.049389567
## [8]  {Y}                 => {Sword} 0.004439512 0.2162162  0.020532741
## [9]  {Black}             => {Sword} 0.003884573 0.3181818  0.012208657
## [10] {Sun}               => {Sword} 0.002774695 0.1000000  0.027746948
## [11] {Pokemon Go}        => {Sword} 0.001664817 0.2142857  0.007769145
## [12] {Shiny Pearl}       => {Sword} 0.001109878 0.1818182  0.006104329
## [13] {1351}              => {Sword} 0.000554939 1.0000000  0.000554939
## [14] {1031}              => {Sword} 0.000554939 1.0000000  0.000554939
## [15] {1026}              => {Sword} 0.000554939 1.0000000  0.000554939
## [16] {947}               => {Sword} 0.000554939 1.0000000  0.000554939
## [17] {1606}              => {Sword} 0.000554939 1.0000000  0.000554939
## [18] {956}               => {Sword} 0.000554939 1.0000000  0.000554939
```

```
## [19] {988}              => {Sword} 0.000554939 1.0000000  0.000554939
## [20] {976}              => {Sword} 0.000554939 1.0000000  0.000554939
##      lift      count
## [1]  1.0000000 243
## [2]  1.1901641  52
## [3]  1.1917989  36
## [4]  1.1573539  27
## [5]  1.1514966  25
## [6]  1.4080325  15
## [7]  0.8332177  10
## [8]  1.6033812   8
## [9]  2.3595211   7
## [10] 0.7415638   5
## [11] 1.5890653   3
## [12] 1.3482978   2
## [13] 7.4156379   1
## [14] 7.4156379   1
## [15] 7.4156379   1
## [16] 7.4156379   1
## [17] 7.4156379   1
## [18] 7.4156379   1
## [19] 7.4156379   1
## [20] 7.4156379   1
```