

Data Mining Project v2

Khalid

```
booking_data <- read.csv("C:\\Users\\Khalid Laptop\\Desktop\\Data Mining Project\\data2\\Passanger_booking_data.csv")
```

We would like to know if we should offer extra baggage for the package we offer

In “length_of_stay column” there are too many variation, the unique values are dividend into 9 groups:

```
unique(booking_data$length_of_stay)
```

```
## [1] 12 19 20 22 31 48 33 30 25 43 24 17 34 18 37 29 155 88
## [19] 27 21 28 47 35 82 26 90 23 84 61 40 96 69 89 58 56 110
## [37] 45 165 111 38 207 91 1 274 44 57 106 68 87 51 196 77 41 278
## [55] 108 180 72 32 62 50 59 125 124 46 55 60 36 65 208 204 83 42
## [73] 85 64 86 54 238 275 39 95 70 93 49 107 121 203 188 78 209 126
## [91] 80 200 255 63 92 118 79 181 76 94 305 149 177 183 162 71 109 52
## [109] 140 66 184 152 53 291 329 75 130 142 175 73 304 186 116 101 223 415
## [127] 312 143 81 144 134 135 117 120 138 153 74 261 103 104 112 157 266 98
## [145] 318 67 273 105 148 102 119 122 603 465 409 128 233 99 113 147 127 97
## [163] 170 156 160 182 115 158 0 357 173 228 205 178 123 352 141 139 129 301
## [181] 176 332 217 358 285 163 359 348 392 179 132 431 236 353 137 146 174 224
## [199] 164 306 252 171 347 185 150 361 189 343 151 133 230 215 256 365 168 335
## [217] 349 355 331 199 254 100 321 245 350 326 351 356 360 262 364 229 193 194
## [235] 131 114 191 235 239 225 190 166 280 136 330 169 345 362 240 145 159 267
## [253] 363 192 161 322 315 289 244 327 308 610 172 260 778 167 226 379 334 313
## [271] 342 284 237 532 513 201 206 221 242 369 290 297 388 282 218 286 271 259
## [289] 510 272 292 197 277 268 435 276 220 338 303 222 187 247 198 195 214 337
## [307] 241 311 279 316 287 478 399 210 341 250 202 573 216 211 773 339 294 154
## [325] 517 462 577 231 293 213 263 2 3 4 5 6
```

```
library(dplyr)
```

```
##  
## Attaching package: 'dplyr'  
  
## The following objects are masked from 'package:stats':  
##  
##     filter, lag  
  
## The following objects are masked from 'package:base':  
##  
##     intersect, setdiff, setequal, union
```

```
booking_data <- booking_data %>% mutate(  
  length_of_stay_group = cut(  
    length_of_stay,  
    breaks = 9,  
    labels = paste0("Group ", 1:9),  
    include.lowest = TRUE)  
)  
  
unique(booking_data$length_of_stay_group)
```

```
## [1] Group 1 Group 2 Group 3 Group 4 Group 5 Group 7 Group 6 Group 8 Group 9  
## 9 Levels: Group 1 Group 2 Group 3 Group 4 Group 5 Group 6 Group 7 ... Group 9
```

```
# load the library needed for this analysis  
library(dplyr)  
library(arules)
```

what is the likelihood for the customer to ask for extra baggage?

```
## Warning: package 'arules' was built under R version 4.4.2
```

```
## Loading required package: Matrix
```

```
##
```

```
## Attaching package: 'arules'
```

```
## The following object is masked from 'package:dplyr':
```

```
##
```

```
##      recode
```

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

```
##
```

```
##      abbreviate, write
```

```
library(arulesViz)
```

```
## Warning: package 'arulesViz' was built under R version 4.4.2
```

```
# make a new data frame containing the column of interest
```

```
transaction_data <- booking_data %>% select(num_passengers, sales_channel, wants_extra_baggage, trip_type, length_of_stay_group)
```

```
#converting the column into factor
```

```
transaction_data$num_passengers <- as.factor(transaction_data$num_passengers)
```

```
transaction_data$sales_channel <- as.factor(transaction_data$sales_channel)
```

```
transaction_data$wants_extra_baggage <- as.factor(transaction_data$wants_extra_baggage)
```

```
transaction_data$trip_type <- as.factor(transaction_data$trip_type)
```

```
transaction_data$length_of_stay_group <- as.factor(transaction_data$length_of_stay_group)
```

```
# check the factor levels for "wants_extra_baggage"
```

```
unique(transaction_data$wants_extra_baggage)
```

```
## [1] 0 1
```

```
## Levels: 0 1
```

```
# convert the level to Yes and No
```

```
transaction_data$wants_extra_baggage <- factor(transaction_data$wants_extra_baggage, levels = c(0, 1), labels = c("No", "Yes"))
```

```
# check the factor levels for "wants_extra_baggage" again
```

```
unique(transaction_data$wants_extra_baggage)
```

```
## [1] No Yes
## Levels: No Yes
```

```
# convert the data frame in transaction format
transactions <- as(transaction_data, "transactions")

# verify the content of the data frame
itemLabels(transactions)
```

```
## [1] "num_passengers=1" "num_passengers=2"
## [3] "num_passengers=3" "num_passengers=4"
## [5] "num_passengers=5" "num_passengers=6"
## [7] "num_passengers=7" "num_passengers=8"
## [9] "num_passengers=9" "sales_channel=Internet"
## [11] "sales_channel=Mobile" "wants_extra_baggage=No"
## [13] "wants_extra_baggage=Yes" "trip_type=CircleTrip"
## [15] "trip_type=OneWay" "trip_type=RoundTrip"
## [17] "length_of_stay_group=Group 1" "length_of_stay_group=Group 2"
## [19] "length_of_stay_group=Group 3" "length_of_stay_group=Group 4"
## [21] "length_of_stay_group=Group 5" "length_of_stay_group=Group 6"
## [23] "length_of_stay_group=Group 7" "length_of_stay_group=Group 8"
## [25] "length_of_stay_group=Group 9"
```

```
# apriori algorithm
rules <- apriori(
  transactions, # the data
  parameter = list(supp = 0.1, conf = 0.1), # parameter
  appearance = list(rhs = c("wants_extra_baggage=Yes", "wants_extra_baggage=No")) #the rhs
)
```

```
## Apriori
##
## Parameter specification:
## confidence minval smax arem aval originalSupport maxtime support minlen
## 0.1 0.1 1 none FALSE TRUE 5 0.1 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: 5000
##
## set item appearances ...[2 item(s)] done [0.00s].
## set transactions ...[25 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [8 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 done [0.00s].
## writing ... [40 rule(s)] done [0.00s].
## creating S4 object ... done [0.00s].
```

```
# inspect and visualize the rules
inspect(head(rules, 10))
```

##	lhs	rhs	support	confidence	coverage	lift	count
## [1]	{}	=> {wants_extra_baggage=No}	0.3312268	0.3312268	1.0000000	1.0000000	16562
## [2]	{}	=> {wants_extra_baggage=Yes}	0.6687732	0.6687732	1.0000000	1.0000000	33440
## [3]	{num_passengers=2}	=> {wants_extra_baggage=Yes}	0.1847726	0.7215714	0.2560698	1.0789477	9239
## [4]	{num_passengers=1}	=> {wants_extra_baggage=No}	0.2360906	0.3765190	0.6270349	1.1367409	11805
## [5]	{sales_channel=Internet}	=> {wants_extra_baggage=No}	0.2859486	0.3221504	0.8876245	0.9725977	14298
## [6]	{length_of_stay_group=Group 1}	=> {wants_extra_baggage=No}	0.3252470	0.3403868	0.9555218	1.0276549	16263
## [7]	{trip_type=RoundTrip}	=> {wants_extra_baggage=No}	0.3284869	0.3318316	0.9899204	1.0018261	16425
## [8]	{num_passengers=1}	=> {wants_extra_baggage=Yes}	0.3909444	0.6234810	0.6270349	0.9322756	19548
## [9]	{sales_channel=Internet}	=> {wants_extra_baggage=Yes}	0.6016759	0.6778496	0.8876245	1.0135717	30085
## [10]	{length_of_stay_group=Group 1}	=> {wants_extra_baggage=Yes}	0.6302748	0.6596132	0.9555218	0.9863032	31515

```
plot(rules, method = "graph")
```



Let see the highest support and confidence value

```
# sort by support
rules_sorted_supp <- sort(rules, by = "support", decreasing = TRUE)
inspect(head(rules_sorted_supp, 5))
```

##	lhs	rhs	support	confidence	coverage	lift	count
## [1]	{}	=> {wants_extra_baggage=Yes}	0.6687732	0.6687732	1.0000000	1.0000000	33440
## [2]	{trip_type=RoundTrip}	=> {wants_extra_baggage=Yes}	0.6614335	0.6681684	0.9899204	0.9990956	33073

```
## [3] {length_of_stay_group=Group 1} => {wants_extra_baggage=Yes} 0.6302748 0.6596132 0.9555218 0.9863032 31515
## [4] {trip_type=RoundTrip,
##     length_of_stay_group=Group 1} => {wants_extra_baggage=Yes} 0.6231951 0.6589482 0.9457422 0.9853088 31161
## [5] {sales_channel=Internet}      => {wants_extra_baggage=Yes} 0.6016759 0.6778496 0.8876245 1.0135717 30085
```

Sort by confidence

```
rules_sorted_conf <- sort(rules, by = "confidence", decreasing = TRUE)
inspect(head(rules_sorted_conf, 5))
```

	lhs	rhs	support	confidence	coverage	lift	count
## [1]	{num_passengers=2, sales_channel=Internet}	=> {wants_extra_baggage=Yes}	0.1664133	0.7282514	0.2285109	1.088936	8321
## [2]	{num_passengers=2, sales_channel=Internet, trip_type=RoundTrip}	=> {wants_extra_baggage=Yes}	0.1645934	0.7279965	0.2260910	1.088555	8230
## [3]	{num_passengers=2, sales_channel=Internet, length_of_stay_group=Group 1}	=> {wants_extra_baggage=Yes}	0.1589936	0.7221364	0.2201712	1.079793	7950
## [4]	{num_passengers=2, sales_channel=Internet, trip_type=RoundTrip, length_of_stay_group=Group 1}	=> {wants_extra_baggage=Yes}	0.1572137	0.7218549	0.2177913	1.079372	7861
## [5]	{num_passengers=2}	=> {wants_extra_baggage=Yes}	0.1847726	0.7215714	0.2560698	1.078948	9239