# Data Mining Project

Hashwineey, Zuhair, Khalid

## Data Cleaning Process

### Load data into R

```
travel <- read.csv('Passanger_booking_data.csv')
```

### Load data into R

```
library(arules)
```

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

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

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

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

```r
library(arulesViz)
```

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

## Looking into the detail of the data

```r
head(travel)
```

```
##   num_passengers sales_channel trip_type purchase_lead length_of_stay
## 1              1      Internet RoundTrip            21             12
## 2              2      Internet RoundTrip           262             19
## 3              1      Internet RoundTrip           112             20
## 4              2      Internet RoundTrip           243             22
## 5              1      Internet RoundTrip            96             31
## 6              2      Internet RoundTrip            68             22
##   flight_hour flight_day  route booking_origin wants_extra_baggage
## 1           6        Tue AKLHGH      Australia                   0
## 2           7        Sat AKLDEL    New Zealand                   1
## 3           3        Sat AKLDEL    New Zealand                   0
## 4          17        Wed AKLDEL          India                   1
## 5           4        Sat AKLDEL    New Zealand                   0
## 6          15        Wed AKLDEL          India                   1
##   wants_preferred_seat wants_in_flight_meals flight_duration booking_complete
## 1                    0                     0           7.21                1
## 2                    0                     0           5.52                0
## 3                    0                     0           5.52                0
## 4                    1                     0           5.52                0
## 5                    0                     1           5.52                0
## 6                    0                     1           5.52                0
```

```r
tail(travel)
```

```
##       num_passengers sales_channel trip_type purchase_lead length_of_stay
## 49997              2      Internet RoundTrip            25              6
## 49998              2      Internet RoundTrip            27              6
```

```
## 49999              1      Internet RoundTrip           111              6
## 50000              1      Internet RoundTrip            24              6
## 50001              1      Internet RoundTrip            15              6
## 50002              1      Internet RoundTrip            19              6
##       flight_hour flight_day  route booking_origin wants_extra_baggage
## 49997           9        Sun PERPNH      Australia                   0
## 49998           9        Sat PERPNH      Australia                   1
## 49999           4        Sun PERPNH      Australia                   0
## 50000          22        Sat PERPNH      Australia                   0
## 50001          11        Mon PERPNH      Australia                   1
## 50002          10        Thu PERPNH      Australia                   0
##       wants_preferred_seat wants_in_flight_meals flight_duration
## 49997                    0                     0           5.62
## 49998                    0                     1           5.62
## 49999                    0                     0           5.62
## 50000                    0                     1           5.62
## 50001                    0                     1           5.62
## 50002                    1                     0           5.62
##       booking_complete
## 49997                0
## 49998                0
## 49999                0
## 50000                0
## 50001                0
## 50002                0
```

```
str(travel)
```

```
## 'data.frame':    50002 obs. of  14 variables:
##  $ num_passengers       : int  1 2 1 2 1 2 1 3 2 1 ...
##  $ sales_channel        : chr  "Internet" "Internet" "Internet" "Internet" ...
##  $ trip_type            : chr  "RoundTrip" "RoundTrip" "RoundTrip" "RoundTrip" ...
##  $ purchase_lead        : int  21 262 112 243 96 68 3 201 238 80 ...
##  $ length_of_stay       : int  12 19 20 22 31 22 48 33 19 22 ...
##  $ flight_hour          : int  6 7 3 17 4 15 20 6 14 4 ...
##  $ flight_day           : chr  "Tue" "Sat" "Sat" "Wed" ...
##  $ route                : chr  "AKLHGH" "AKLDEL" "AKLDEL" "AKLDEL" ...
##  $ booking_origin       : chr  "Australia" "New Zealand" "New Zealand" "India" ...
##  $ wants_extra_baggage  : int  0 1 0 1 0 1 1 1 1 0 ...
```

```
##  $ wants_preferred_seat : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ wants_in_flight_meals: int  0 0 0 0 1 1 1 1 1 1 ...
##  $ flight_duration      : num  7.21 5.52 5.52 5.52 5.52 5.52 5.52 5.52 5.52 5.52 ...
##  $ booking_complete     : int  1 0 0 0 0 0 0 0 0 0 ...
```

```r
dim(travel)
```

```
## [1] 50002    14
```

### The collumn of interest

```r
# num_passengers, sales_channel, trip_type, purchase_lead
# booking_origin, wants_extra_baggage, wants_preferred_seat, wants_in_flight_meals
# flight_duration, booking_complete

##  Total 10 columns
```

### Cleaning process on the collumn

```r
# changing the value for sales_channel from "Internet" and "Mobile" into "Website" and "Mobile_App" respectively
travel$sales_channel[travel$sales_channel %in% "Internet"] = "Website"
travel$sales_channel[travel$sales_channel %in% "Mobile"] = "Mobile_App"

# grouping the value in the flight_duration in 5 groups
travel$flight_duration <- cut(travel$flight_duration, breaks = 5, labels = c(1, 2, 3, 4, 5))

# grouping the value in the purchase_lead in 3 groups
travel$purchase_lead <- cut(travel$purchase_lead, breaks = 3, labels = c(1, 2, 3))

# changing the binary collumn into Yes and No respectively
columns_to_convert <- c("wants_extra_baggage", "wants_preferred_seat", "wants_in_flight_meals", "booking_complete")
travel[columns_to_convert] <- lapply(travel[columns_to_convert], function(x) ifelse(x == 1, "Yes", "No"))
```

### Compile all of the column into a new dataset

```
travel_a <- travel[, c("num_passengers", "sales_channel", "trip_type", "purchase_lead",
                       "booking_origin", "wants_extra_baggage", "wants_preferred_seat",
                       "wants_in_flight_meals", "flight_duration", "booking_complete")]
```

### Preparing dataset to be converted into transaction data

```
# converting character columns to factors
change_into_factor <- c("num_passengers", "sales_channel", "trip_type", "booking_origin",
                    "wants_extra_baggage", "wants_preferred_seat",
                    "wants_in_flight_meals", "booking_complete")
travel_a[change_into_factor] <- lapply(travel_a[change_into_factor], as.factor)

# converting dataset into transaction format
transactions <- as(travel_a, "transactions")
```

### Creating Association Rule based on the question of interest

**1. Between Website and Mobile_App, which one have the highest confidence level for booking_complete as Yes?**

```
# 1. Between Website and Mobile_App, which one have the highest confidence level for booking_complete as Yes?
## Generate rules for Website
rules_website <- apriori(transactions,
                         parameter = list(supp = 0.01, conf = 0.1),
                         appearance = list(lhs = c("sales_channel=Website"),
                                           rhs = c("booking_complete=Yes")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[2 item(s)] done [0.00s].
## set transactions ...[2 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [2 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```r
# Generate rules for Mobile_App
rules_mobile_app <- apriori(transactions,
                            parameter = list(supp = 0.01, conf = 0.1),
                            appearance = list(lhs = c("sales_channel=Mobile_App"),
                                              rhs = c("booking_complete=Yes")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[2 item(s)] done [0.00s].
## set transactions ...[2 item(s), 50002 transaction(s)] done [0.01s].
```

```
## sorting and recoding items ... [2 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# inspect the value
inspect(rules_website)
```

```
##     lhs                         rhs                     support   confidence
## [1] {}                       => {booking_complete=Yes} 0.1495740 0.149574
## [2] {sales_channel=Website} => {booking_complete=Yes} 0.1373945 0.154789
##     coverage  lift     count
## [1] 1.0000000 1.000000 7479
## [2] 0.8876245 1.034866 6870
```

```
inspect(rules_mobile_app)
```

```
##     lhs                            rhs                     support    confidence
## [1] {}                          => {booking_complete=Yes} 0.14957402 0.1495740
## [2] {sales_channel=Mobile_App} => {booking_complete=Yes} 0.01217951 0.1083823
##     coverage  lift      count
## [1] 1.0000000 1.0000000 7479
## [2] 0.1123755 0.7246063  609
```

**2. What is the likelihood for customer that do longer flight duration to asked for preferred seat and in flight meals?**

```
# Generate rules for flight_duration = 1 (shortest flight duration group)
rules_flight_duration_1 <- apriori(transactions,
                                   parameter = list(supp = 0.01, conf = 0.1),
                                   appearance = list(lhs = c("flight_duration=1"),
                                                     rhs = c("wants_preferred_seat=Yes", "wants_preferred_seat=No",
                                                             "wants_in_flight_meals=Yes", "wants_in_flight_meals=No")))
```

```
## Apriori
```

```
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[5 item(s)] done [0.00s].
## set transactions ...[5 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [5 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [8 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# Generate rules for flight_duration = 5 (longest flight duration group)
rules_flight_duration_5 <- apriori(transactions,
                                   parameter = list(supp = 0.01, conf = 0.1),
                                   appearance = list(lhs = c("flight_duration=5"),
                                                     rhs = c("wants_preferred_seat=Yes", "wants_preferred_seat=No",
                                                             "wants_in_flight_meals=Yes", "wants_in_flight_meals=No")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[5 item(s)] done [0.00s].
## set transactions ...[5 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [5 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [8 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# Inspect the rules
inspect(sort(rules_flight_duration_1, by = "confidence", decreasing = TRUE))
```

```
##     lhs                    rhs                          support    confidence
## [1] {flight_duration=1} => {wants_preferred_seat=No}   0.19965201 0.7517319
## [2] {}                  => {wants_preferred_seat=No}   0.70303188 0.7030319
## [3] {flight_duration=1} => {wants_in_flight_meals=No}  0.17323307 0.6522590
## [4] {}                  => {wants_in_flight_meals=No}  0.57285709 0.5728571
## [5] {}                  => {wants_in_flight_meals=Yes} 0.42714291 0.4271429
## [6] {flight_duration=1} => {wants_in_flight_meals=Yes} 0.09235631 0.3477410
## [7] {}                  => {wants_preferred_seat=Yes}  0.29696812 0.2969681
## [8] {flight_duration=1} => {wants_preferred_seat=Yes}  0.06593736 0.2482681
##     coverage  lift      count
## [1] 0.2655894 1.0692715  9983
## [2] 1.0000000 1.0000000 35153
## [3] 0.2655894 1.1386069  8662
## [4] 1.0000000 1.0000000 28644
## [5] 1.0000000 1.0000000 21358
## [6] 0.2655894 0.8141092  4618
## [7] 1.0000000 1.0000000 14849
## [8] 0.2655894 0.8360092  3297
```

```
inspect(sort(rules_flight_duration_5, by = "confidence", decreasing = TRUE))
```

```
##     lhs                  rhs                          support   confidence
## [1] {}                => {wants_preferred_seat=No}   0.7030319 0.7030319
```

```
## [2] {flight_duration=5} => {wants_preferred_seat=No}   0.2945082 0.6595011
## [3] {}                  => {wants_in_flight_meals=No}  0.5728571 0.5728571
## [4] {flight_duration=5} => {wants_in_flight_meals=Yes} 0.2301108 0.5152940
## [5] {flight_duration=5} => {wants_in_flight_meals=No}  0.2164513 0.4847060
## [6] {}                  => {wants_in_flight_meals=Yes} 0.4271429 0.4271429
## [7] {flight_duration=5} => {wants_preferred_seat=Yes}  0.1520539 0.3404989
## [8] {}                  => {wants_preferred_seat=Yes}  0.2969681 0.2969681
##     coverage  lift      count
## [1] 1.0000000 1.0000000 35153
## [2] 0.4465621 0.9380814 14726
## [3] 1.0000000 1.0000000 28644
## [4] 0.4465621 1.2063738 11506
## [5] 0.4465621 0.8461203 10823
## [6] 1.0000000 1.0000000 21358
## [7] 0.4465621 1.1465840  7603
## [8] 1.0000000 1.0000000 14849
```

**3. Does customer with shorter purchase lead will complete their booking?**

```r
# Generate rules for purchase_lead = 1
rules_purchase_lead_1 <- apriori(transactions,
                                 parameter = list(supp = 0.01, conf = 0.1),
                                 appearance = list(lhs = c("purchase_lead=1"),
                                                   rhs = c("booking_complete=Yes", "booking_complete=No")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[3 item(s)] done [0.00s].
## set transactions ...[3 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [3 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [4 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```r
# Generate rules for purchase_lead = 2
rules_purchase_lead_2 <- apriori(transactions,
                                 parameter = list(supp = 0.01, conf = 0.1),
                                 appearance = list(lhs = c("purchase_lead=2"),
                                                   rhs = c("booking_complete=Yes", "booking_complete=No")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      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: 500
##
## set item appearances ...[3 item(s)] done [0.00s].
## set transactions ...[3 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [3 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [3 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```r
# Generate rules for purchase_lead = 3
rules_purchase_lead_3 <- apriori(transactions,
                                 parameter = list(supp = 0.00001, conf = 0.1),
                                 appearance = list(lhs = c("purchase_lead=3"),
                                                   rhs = c("booking_complete=Yes", "booking_complete=No")))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5   1e-05      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: 0
##
## set item appearances ...[3 item(s)] done [0.00s].
## set transactions ...[3 item(s), 50002 transaction(s)] done [0.01s].
## sorting and recoding items ... [3 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [4 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```r
# Inspect the value
inspect(sort(rules_purchase_lead_1, by = "confidence", decreasing = TRUE))
```

```
##     lhs                 rhs                    support   confidence coverage
## [1] {}                => {booking_complete=No}  0.8504260 0.8504260  1.000000
## [2] {purchase_lead=1} => {booking_complete=No}  0.8083277 0.8498318  0.951162
## [3] {purchase_lead=1} => {booking_complete=Yes} 0.1428343 0.1501682  0.951162
## [4] {}                => {booking_complete=Yes} 0.1495740 0.1495740  1.000000
##     lift      count
## [1] 1.0000000 42523
```

```
## [2] 0.9993013 40418
## [3] 1.0039726  7142
## [4] 1.0000000  7479
```

```
inspect(sort(rules_purchase_lead_2, by = "confidence", decreasing = TRUE))
```

```
##     lhs                  rhs                    support    confidence
## [1] {purchase_lead=2} => {booking_complete=No}  0.04195832 0.8623099
## [2] {}                => {booking_complete=No}  0.85042598 0.8504260
## [3] {}                => {booking_complete=Yes} 0.14957402 0.1495740
##     coverage   lift     count
## [1] 0.04865805 1.013974  2098
## [2] 1.00000000 1.000000 42523
## [3] 1.00000000 1.000000  7479
```

```
inspect(sort(rules_purchase_lead_3, by = "confidence", decreasing = TRUE))
```

```
##     lhs                  rhs                    support      confidence
## [1] {}                => {booking_complete=No}  0.8504259830 0.8504260
## [2] {purchase_lead=3} => {booking_complete=No}  0.0001399944 0.7777778
## [3] {purchase_lead=3} => {booking_complete=Yes} 0.0000399984 0.2222222
## [4] {}                => {booking_complete=Yes} 0.1495740170 0.1495740
##     coverage     lift      count
## [1] 1.0000000000 1.0000000 42523
## [2] 0.0001799928 0.9145743     7
## [3] 0.0001799928 1.4857007     2
## [4] 1.0000000000 1.0000000  7479
```