Data Preparation

Chindu

Cleaning/Feature engineering/Data manipulation

A sample data of a travel industry is prepared for modelling purpose. The data set consists of 999 observations and 23 variables about customer enquiries about holiday packages.This is a randomly fabricated dataset just for the purpose of demonstracting some of the critical steps in data preparation phase.This dataset is prepared to create predictive models for predicting the 'Booked.Status'.'Data for cleaning.csv' is the input file and 'ReadyforModelling.csv' is the output file after the data preperation methods are carried out.

List of library used: library(lubridate), library(zoo), library(imputeTS), library(DataExplorer), library(data.table)

Read data and get a basic understanding of the data

```
data<- read.csv("data for cleaning.csv")
head(data)
```

```
##   Enquiry.Date Enquiry.Time Allocated.Time Web.or.Phone
## 1     1/1/2017     14:52:40 Extremely Fast        PHONE
## 2     1/1/2017     12:00:54 Extremely Fast        PHONE
## 3     1/1/2017     17:25:01 Extremely Fast        PHONE
## 4     1/1/2017      8:46:38           Slow          WEB
## 5     1/1/2017      8:38:38           Slow          WEB
## 6     1/2/2017     23:38:38 Extremely Fast          WEB
##   Answered.by.specialist ConversationRCD TempSent Holiday.Type Accom.type
## 1                                     14        1            A     grade1
## 2                                     18        1            C     grade1
## 3                                      3        2            A     grade1
## 4                                      9        1            A     grade1
## 5                                     15        4            A     grade1
## 6                    Yes               6        3            B     grade1
##   Dep.Airport   Dep.Date Lead.Time Destination Duration Adults Children
## 1     Lon All 12/19/2017        50     JH Area       14      6        2
## 2 Any Airport  4/10/2017        14          AB       10      2        2
## 3          GG 10/14/2017        40          AC       14      4        1
## 4         MCH   4/8/2017        13          AB       14      2        1
## 5     Lon Gat   6/7/2018        74          AC       14      7        1
## 6     Lon Gat  4/11/2018        66          AB       14      6        2
##   Infants Transport.Type Answered.Q Notes.Completed Title Enquiry.Comments
## 1       0              A        YES              NO   Mrs               NO
## 2       0  None Required         NO              NO   Mrs               NO
## 3       0              A         NO              NO   Mrs               NO
## 4       0              A        YES              NO   Mrs               NO
## 5       0              A        YES              NO    Mr               NO
## 6       0              A        YES              NO   Mrs               NO
##   Booked.Status
## 1           YES
## 2           YES
## 3           YES
## 4            No
## 5            No
## 6            No
```

Check the structure of the dataset

```
str(data)
```

```
## 'data.frame':    999 obs. of  23 variables:
##  $ Enquiry.Date          : Factor w/ 501 levels "1/1/2017","1/1/2018",..: 1 1 1 1 1 22 41 41 41 47 .
##  $ Enquiry.Time          : Factor w/ 754 levels "0:02:38","0:10",..: 343 149 430 678 667 522 13 341 4
##  $ Allocated.Time        : Factor w/ 3 levels "Extremely Fast",..: 1 1 1 3 3 1 1 3 3 1 ...
##  $ Web.or.Phone          : Factor w/ 2 levels "PHONE","WEB": 1 1 1 2 2 2 2 2 2 1 ...
##  $ Answered.by.specialist: Factor w/ 2 levels "","Yes": 1 1 1 1 1 2 2 1 1 1 ...
##  $ ConversationRCD       : int  14 18 3 9 15 6 0 9 10 25 ...
##  $ TempSent              : int  1 1 2 1 4 3 1 1 4 8 ...
##  $ Holiday.Type          : Factor w/ 6 levels "A","B","C","D",..: 1 3 1 1 1 2 1 2 1 2 ...
##  $ Accom.type            : Factor w/ 5 levels "","grade1","grade2",..: 2 2 2 2 2 2 3 2 3 3 ...
##  $ Dep.Airport           : Factor w/ 17 levels "AD","Any Airport",..: 12 2 10 16 13 13 12 12 12 12 .
##  $ Dep.Date              : Factor w/ 550 levels "1/1/2018","1/11/2018",..: 127 208 22 253 343 210 45
##  $ Lead.Time             : int  50 14 40 13 74 66 42 94 39 50 ...
##  $ Destination           : Factor w/ 35 levels " AA Resort"," AB",..: 17 2 3 2 3 2 2 9 3 3 ...
##  $ Duration              : int  14 10 14 14 14 14 10 10 14 13 ...
##  $ Adults                : int  6 2 4 2 7 6 2 9 3 2 ...
##  $ Children              : int  2 2 1 1 1 2 0 2 1 2 ...
##  $ Infants               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Transport.Type        : Factor w/ 4 levels "","A","B","None Required": 2 4 2 2 2 2 4 3 3 3 ...
##  $ Answered.Q            : Factor w/ 2 levels "NO","YES": 2 1 1 2 2 2 1 1 2 2 ...
##  $ Notes.Completed       : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Title                 : Factor w/ 5 levels "Dr","Miss","Mr",..: 4 4 4 4 3 4 4 3 2 4 ...
##  $ Enquiry.Comments      : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Booked.Status         : Factor w/ 2 levels "No","YES": 2 2 2 1 1 1 1 1 1 2 ...
```

From understanding the structure, it is observed that variables such as Dep.Date,Enquiry.Date and Enquiry.Time have significant number of levels. It would be ideal to categorise them into larger groups. For example Dep.Date can be categorised into months or seasons, such analysis would allow us to get better insights from the data. It is also critical to check the structure in which R has identified each variable (factor, numerical, integet, etc). In this dataset, R has identified dates as factor. This should be coverted to date format.

Before further analysis, it is a good practice to eliminate variables which are not relevant to the analysis, in this case ConversationRCD as well as TempSent will be eliminated.

```
data$TempSent<-NULL
data$ConversationRCD<-NULL
```

Convert Dep.Date and Enquiry.Date to date format

```
data$Enquiry.Date<- as.character(data$Enquiry.Date)
data$Enquiry.Date<-mdy(data$Enquiry.Date)
data$Dep.Date<- as.character(data$Dep.Date)
data$Dep.Date<-mdy(data$Dep.Date)
```

Apply feature engineering to create various date related columns which might give us better insights

```
data$EnquiryYear<-factor(year(data$Enquiry.Date))
data$EnquiryMonth<-factor(month(data$Enquiry.Date))
data$EnquiryDay<-day(data$Enquiry.Date)
data$EnquiryWeekday<-factor(weekdays(data$Enquiry.Date))
data$DepYear<-factor(year(data$Dep.Date))
data$DepMonth<-factor(month(data$Dep.Date))
data$DepDay<-day(data$Dep.Date)
data$DepWeekday<-factor(weekdays(data$Dep.Date))
```

Change Enquiry.time to various time related levels to give us better insights

```
data$Enquiry.Time <- as.numeric(gsub("\\:.*$", "", data$Enquiry.Time))
data$Enquiry.Timecat<-ifelse(data$Enquiry.Time>=9 &
                               data$Enquiry.Time<=21,"Business_Hour","Closed")
data$Enquiry.Timecat<-factor(data$Enquiry.Timecat)

data$Enquiry.Time_class <- with(data,  ifelse(Enquiry.Time >= 6 &
                                                Enquiry.Time<=12, "morning",
                                       ifelse(Enquiry.Time>12 &
                                                Enquiry.Time<=18, "afternoon", "night")))
data$Enquiry.Time<- NULL
data$Enquiry.Time_class<-factor(data$Enquiry.Time_class)
```

Change Dep.Date to seasons this could give a better idea of popular destinations for each seasons

```
yq <- as.yearqtr(as.yearmon(data$Dep.Date, "%m/%d/%Y") + 1/12)
data$DepartureSeason <- factor(format(yq, "%q"), levels = 1:4,
                               labels = c("winter", "spring", "summer", "fall"))
```

Since Enquiry.Date and Dep.Date has no further use in this analysis, these variables are removed

```
data$Enquiry.Date<-NULL
data$Dep.Date<-NULL
```

Check for missing values

```
colSums(is.na(data))
```

```
##         Allocated.Time           Web.or.Phone Answered.by.specialist
##                      0                      0                      0
##           Holiday.Type             Accom.type            Dep.Airport
##                      0                      0                      0
##              Lead.Time            Destination               Duration
##                      0                      0                     17
##                 Adults               Children                Infants
##                      0                      0                      0
##         Transport.Type             Answered.Q        Notes.Completed
##                      0                      0                      0
##                  Title       Enquiry.Comments          Booked.Status
##                      0                      0                      0
##            EnquiryYear           EnquiryMonth             EnquiryDay
##                      0                      0                      0
##         EnquiryWeekday                DepYear               DepMonth
##                      0                      0                      0
##                 DepDay             DepWeekday        Enquiry.Timecat
##                      0                      0                      0
##     Enquiry.Time_class        DepartureSeason
##                      0                      0
```

The variable Duration has missing values. Since only a small number of observations have missing values, it was decided that the missing values will be replace by the median value

```
data$Duration<-na.mean(data$Duration,option="median")
```

To further understand the data, the summary function is used

```
summary(data)
```

```
##         Allocated.Time Web.or.Phone Answered.by.specialist  Holiday.Type
##   Extremely Fast:259    PHONE:197       :490                A      :684
##   Fast          :135    WEB  :802    Yes:509                B      :136
##   Slow          :605                                        C      : 28
##                                                             D      : 34
##                                                             E      :115
##                                                             RV Tour:  2
##
##    Accom.type        Dep.Airport    Lead.Time          Destination
##         : 91   MCH          :314   Min.   :-10.00    AC       :323
##   grade1:379   Lon All      :289   1st Qu.: 30.00    AB       :233
##   grade2:476   Lon Gat      :141   Median : 48.00    JH Area  : 95
##   grade3: 52   GG           : 66   Mean   : 50.47    AA Resort: 91
##   None  :  1   Any Airport : 42   3rd Qu.: 67.00    DC Drive : 51
##                Lon Heathrow: 40   Max.   :140.00    CC City  : 42
##                (Other)     :107                    (Other)  :164
##     Duration         Adults         Children        Infants
##   Min.   : 1.00   Min.   : 1.00   Min.   :0.000   Min.   :  0.0000
##  1st Qu.:14.00   1st Qu.: 2.00   1st Qu.:0.000   1st Qu.:  0.0000
##  Median :14.00   Median : 3.00   Median :1.000   Median :  0.0000
##  Mean   :13.48   Mean   : 3.63   Mean   :0.955   Mean   :  0.2923
##  3rd Qu.:14.00   3rd Qu.: 4.00   3rd Qu.:2.000   3rd Qu.:  0.0000
##  Max.   :28.00   Max.   :18.00   Max.   :6.000   Max.   :255.0000
##
##        Transport.Type Answered.Q Notes.Completed  Title
##                :  5    NO :486    NO :712         Dr  :  4
##  A            :518    YES:513    YES:287         Miss:126
##  B            :252                               Mr  :406
##  None Required:224                               Mrs :414
##                                                  Ms  : 49
##
##
##  Enquiry.Comments Booked.Status EnquiryYear  EnquiryMonth   EnquiryDay
##  NO :751          No :750       2017:484    1      :150   Min.   : 1.00
##  YES:248          YES:249       2018:515    4      :101   1st Qu.: 8.00
##                                             2      :100   Median :16.00
##                                             5      :100   Mean   :15.77
##                                             9      : 94   3rd Qu.:24.00
##                                             7      : 80   Max.   :31.00
##                                             (Other):374
##    EnquiryWeekday DepYear       DepMonth       DepDay          DepWeekday
##  Friday   :107    2017:122   8      :246   Min.   : 1.00   Friday   :136
##  Monday   :151    2018:415   10     :139   1st Qu.: 7.00   Monday   :157
##  Saturday :122    2019:390   7      :124   Median :15.00   Saturday :182
##  Sunday   :218    2020: 71   4      : 91   Mean   :15.09   Sunday   : 94
##  Thursday :119    2021:  1   5      : 88   3rd Qu.:22.00   Thursday :125
##  Tuesday  :136               9      : 87   Max.   :31.00   Tuesday  :141
##  Wednesday:146               (Other):224                   Wednesday:164
##       Enquiry.Timecat Enquiry.Time_class DepartureSeason
##  Business_Hour:748    afternoon:317      winter: 76
##  Closed       :251    morning  :581      spring:231
##                       night    :101      summer:426
##                                          fall  :266
##
##
##
```

The summary function shows the statistics of the numerical variables and the breakdown of the different levels of the categorical variables. The information gained from this function is critical in preparing the data for analysis.

The variable Answered.by.specialist has 490 unlabeled data and 509 labeled as ‘Yes’. This means that only when the event occurs, it was recorded as ‘Yes’ otherwise left blank. These unlabeled observations should be converted to ‘NO’ before further analysis.

```r
data$Answered.by.specialist<-ifelse(data$Answered.by.specialist %in% 'Yes',"1","0")
data$Answered.by.specialist<-factor(data$Answered.by.specialist)
```

From the summary analysis carried out earlier, it is understood that there are some errors in the data. To remove these errors, a function is created. This function converts any values stated by the user to ‘NA’.

```
outlierReplace = function(dataframe, cols, rows, newValue = NA)
{
  if (any(rows))
  {
    set(dataframe, rows, cols, newValue)
  }
}
```

From the understanding of the dataset, Lead.Time refers to the duration before the Dep.Date that the customer has made the enquiry. Based on this knowledge this variable should not have negative values. Hence the outlierfunction is used to eliminate any negative values.

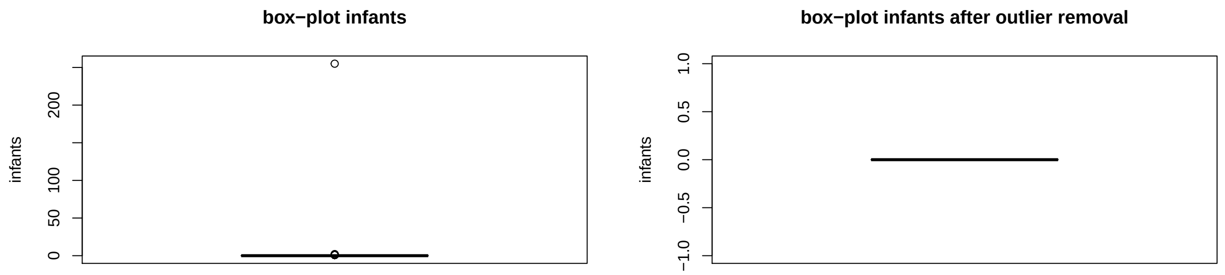
```
outlierReplace(data, "Lead.Time",which(data$Lead.Time<0), NA)
```

Remove all NA values

```
data<-na.omit(data)
```

The variable infants has a maximum value of 255, this is likely to be an error based on the mean and median. Furthermore it is unlikely to have 255 infants in a holiday.To verify the error a box-plot is used to get a better understanding.

```
boxplot(data$Infants, main= 'box-plot infants',ylab='infants')
outliers0 <- boxplot(data$Infants, plot=FALSE)$out
data <- data[-which(data$Infants %in% outliers0),]
boxplot(data$Infants,main='box-plot infants after outlier removal',ylab='infants')
```

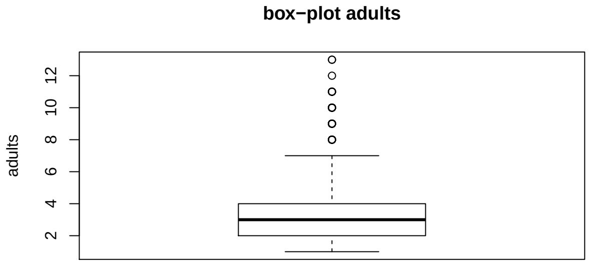


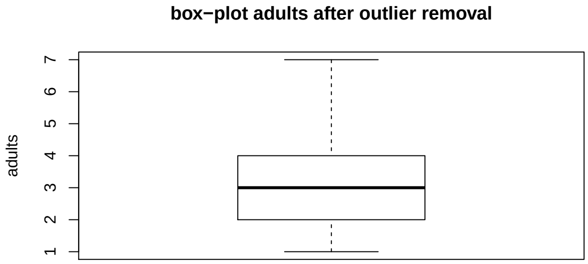
After removing the outliers, the data only contains observaions with 0 infants. Hence, it would not be of any use in the analysis as all cases contains 0 infants. The variable Infants is removed from the dataframe.

```
data$Infants<-NULL
```

From the summary statistics of Adults, the maximum value of adults is far greater than the mean and median value. To better understand this, a scatterplot for adults is created.

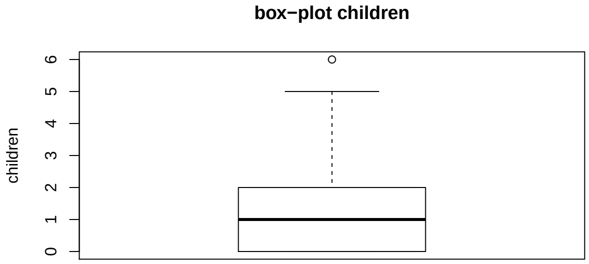
```
boxplot(data$Adults,main='box-plot adults',ylab='adults')
outlierReplace(data,"Adults",which(data$Adults>10),NA)
outliers1 <- boxplot(data$Adults, plot=FALSE)$out
data <- data[-which(data$Adults %in% outliers1),]
boxplot(data$Adults,main='box-plot adults after outlier removal',ylab='adults')
```

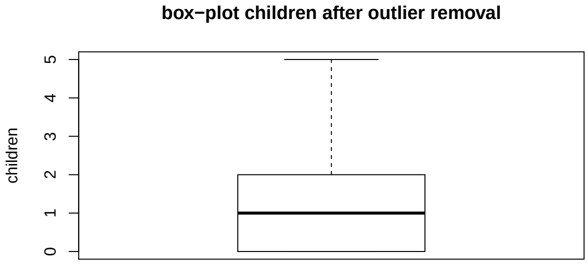




A box plot method was used to deal with the outliers for the valiable 'Children'

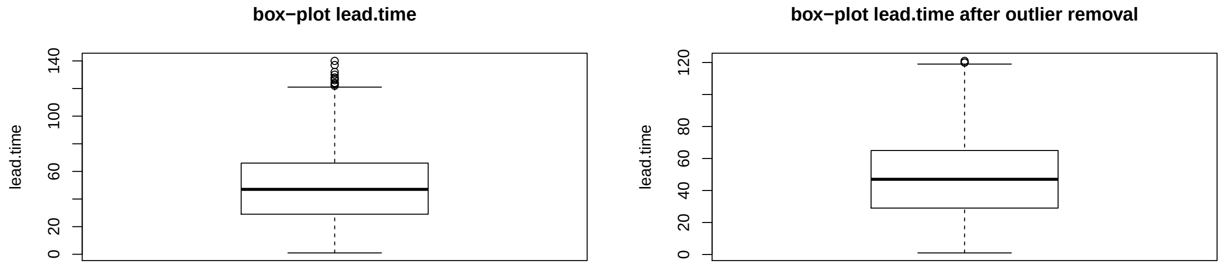
```
boxplot(data$Children,main='box-plot children',ylab='children')
outliers3 <- boxplot(data$Children, plot=FALSE)$out
data <- data[-which(data$Children %in% outliers3),]
boxplot(data$Children, main='box-plot children after outlier removal',ylab='children')
```





Similarly outliers in Lead.Time was treated using the same method

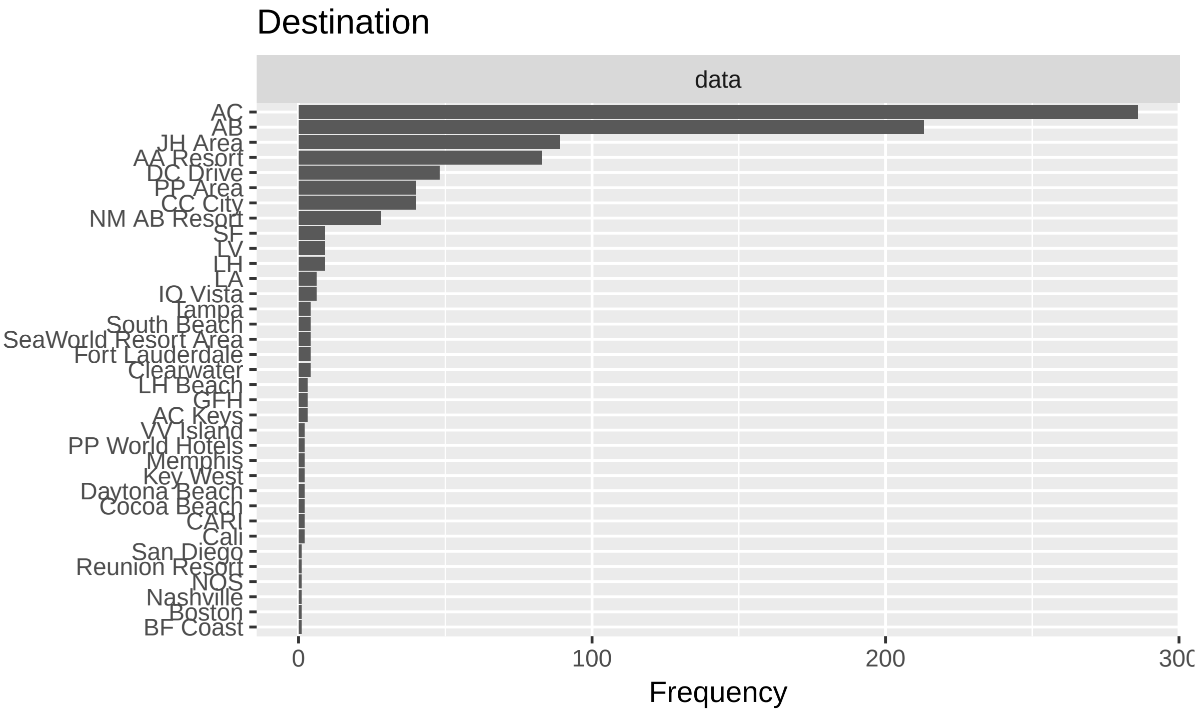
```
boxplot(data$Lead.Time,main='box-plot lead.time',ylab='lead.time')
outliers4 <- boxplot(data$Lead.Time, plot=FALSE)$out
data <- data[-which(data$Lead.Time %in% outliers4),]
boxplot(data$Lead.Time, main='box-plot lead.time after outlier removal',ylab='lead.time')
```



To avoid redundant levels in a categorical variable and to deal with rare levels, we can simply combine the rare levels.In this analysis, combining levels is based on frequency destribution (combine levels having frequency of less than 5%). From the summary statistic, Destination and Dep.Airport has more than 10 levels. A histogram plot is created to understand the levels in these variables and rare levels of these variables are combined.

Destination

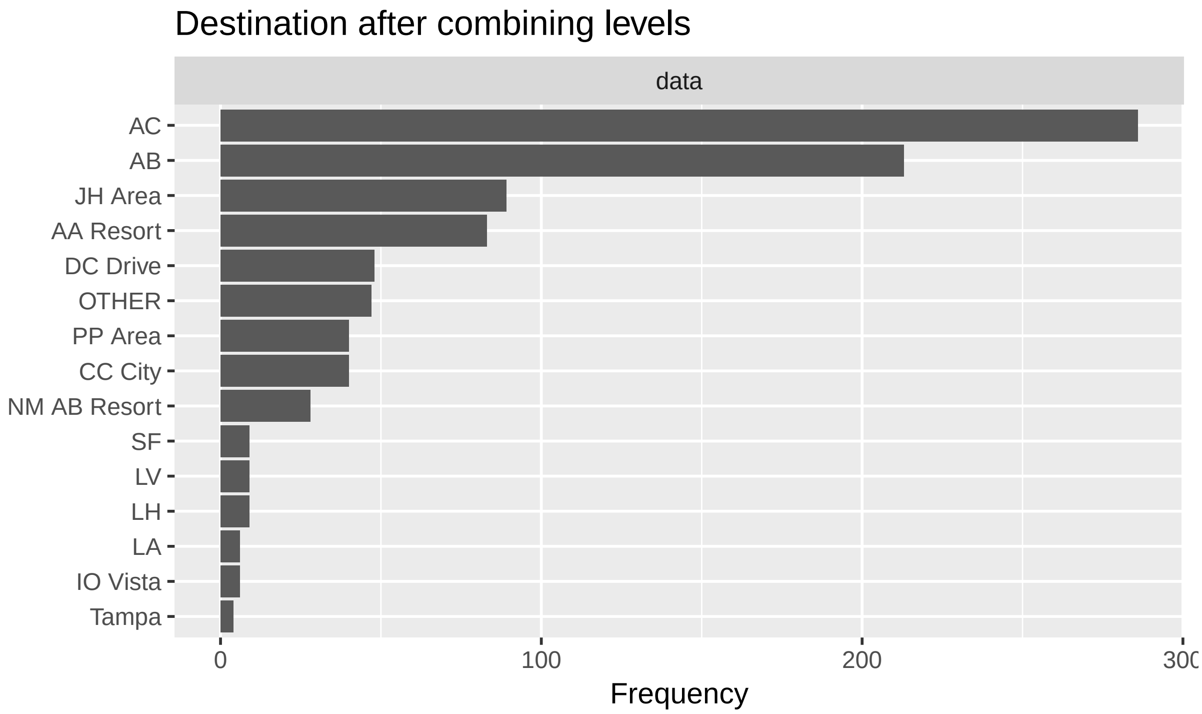
```r
plot_bar(data$Destination,title="Destination")
```



```r
data<-group_category(data=data, feature = "Destination", threshold=0.05, update=TRUE)
data$Destination<- factor(data$Destination)
```

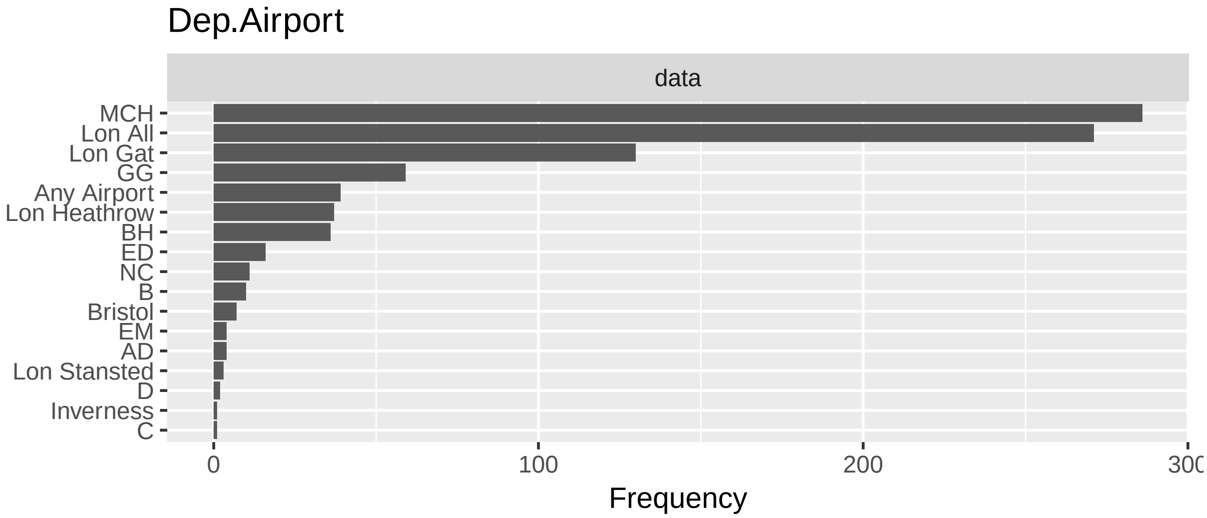
New levels for Destination after combining rare levels

```
plot_bar(data$Destination,title="Destination after combining levels")
```



Dep.Airport

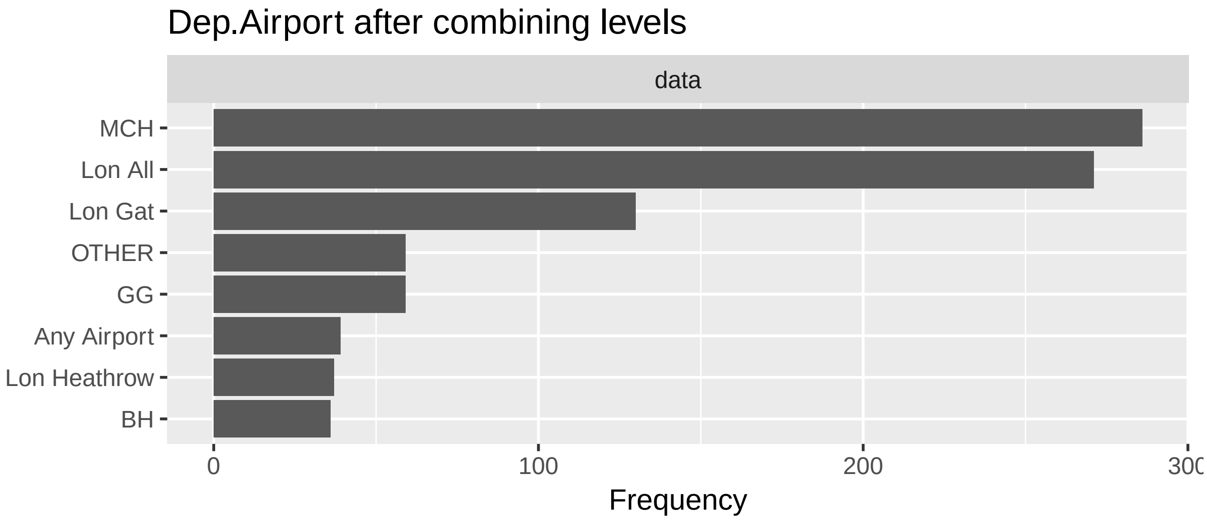
```
plot_bar(data$Dep.Airport,title="Dep.Airport")
```



```
data<-group_category(data=data, feature = "Dep.Airport", threshold=0.05,update=TRUE)
data$Dep.Airport<-factor(data$Dep.Airport)
```

New levels for Dep.Airport after combining rare levels

```
plot_bar(data$Dep.Airport,title="Dep.Airport after combining levels")
```

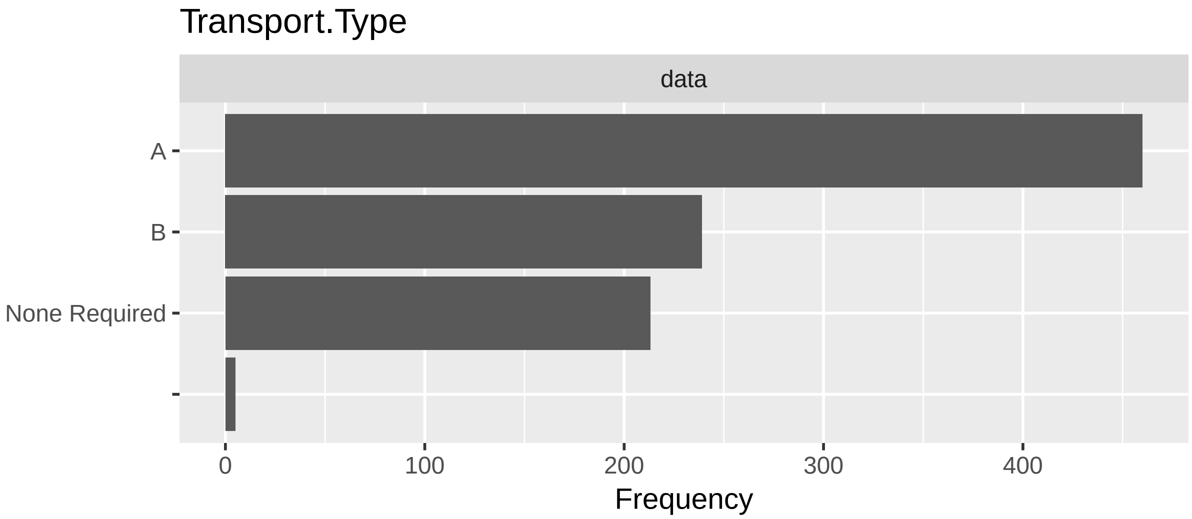


Combine levels based on business logic

Combine unlabbeled points in Transport.Type into 'None'

From the plot of Transport.Type it is identified that some of the points are unlabeled, we can treat the unlabelled points as 'None Required'Combining unlabed points with the the 'None required' level

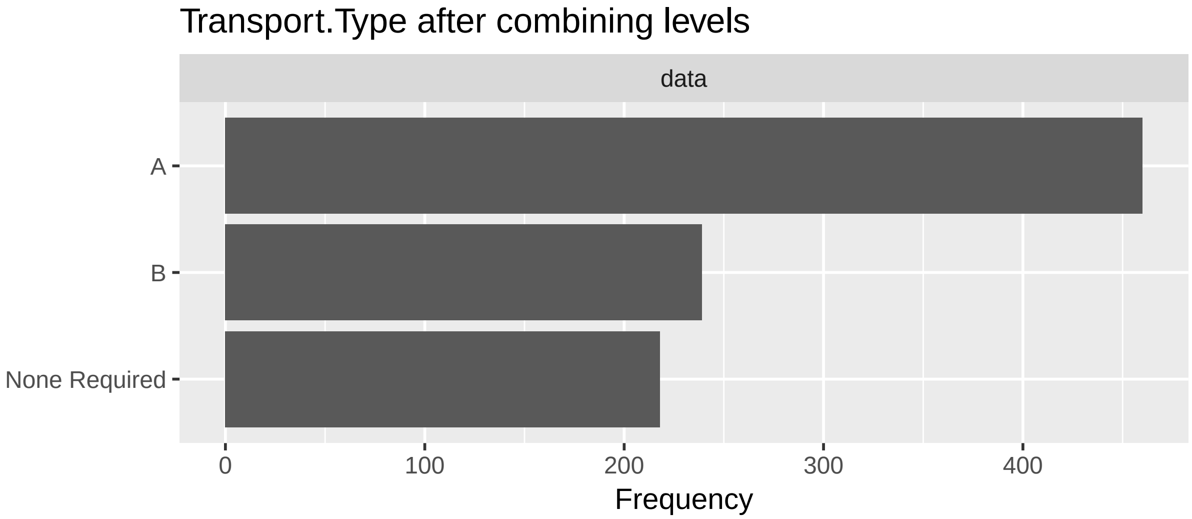
```
plot_bar(data$Transport.Type,title="Transport.Type")
```



```
data$Transport.Type <- with(data,  ifelse(Transport.Type %in% "A","A",
                                         ifelse(Transport.Type %in% "B", "B", "None Required")))
data$Transport.Type<- factor(data$Transport.Type)
```

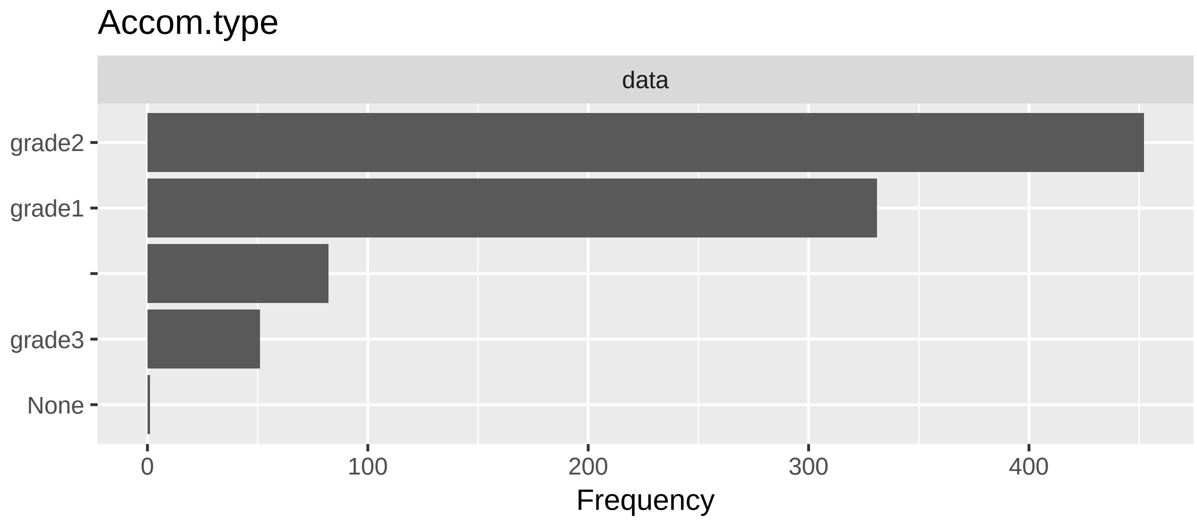
New levels for Transport.Type after combining rare levels

```
plot_bar(data$Transport.Type,title="Transport.Type after combining levels")
```



Combine unlabbeled points in Accom.type into 'None'

```
plot_bar(data$Accom.type,title="Accom.type")
```

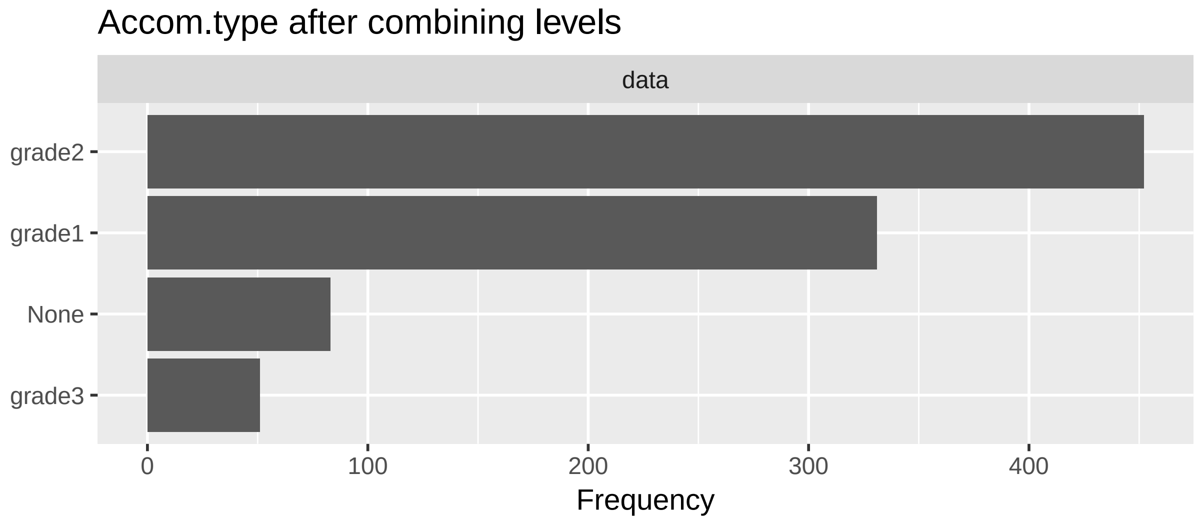


```
data$Accom.type <- with(data,  ifelse(Accom.type %in% "grade2","grade2",
                                 ifelse(Accom.type %in% "grade1", "grade1",
                                    ifelse(Accom.type %in% "grade3", "grade3","None"))))

data$Accom.type<- factor(data$Accom.type)
```

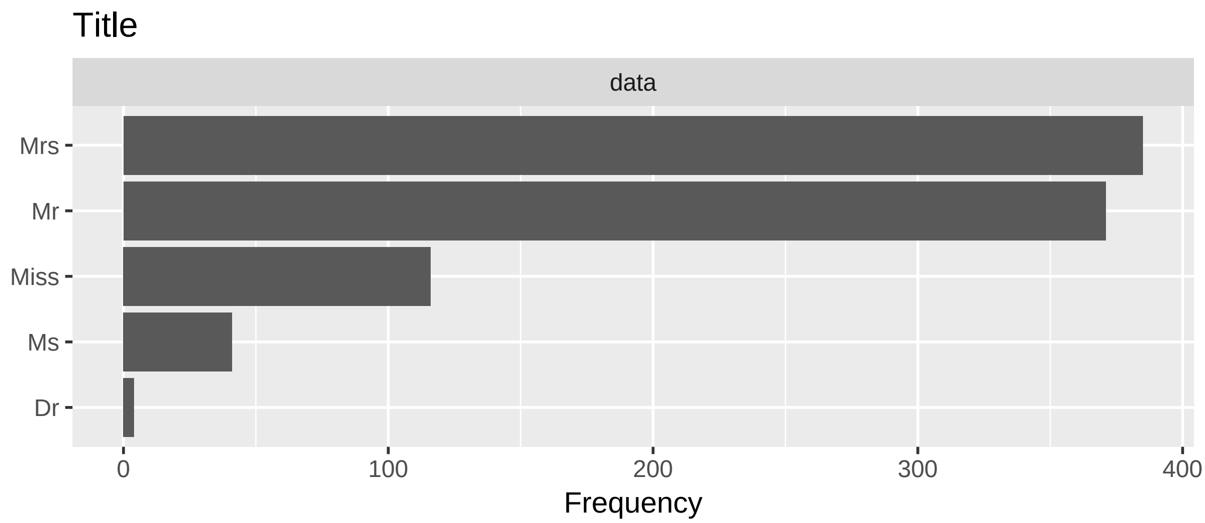
New levels for Accom.type after combining rare levels

```
plot_bar(data$Accom.type,title="Accom.type after combining levels")
```



It could be ideal to analyse based on gender than based on Title, converting title to M for male and F for female. An assumption is made that "Dr" refers to male

```
plot_bar(data$Title, title="Title")
```

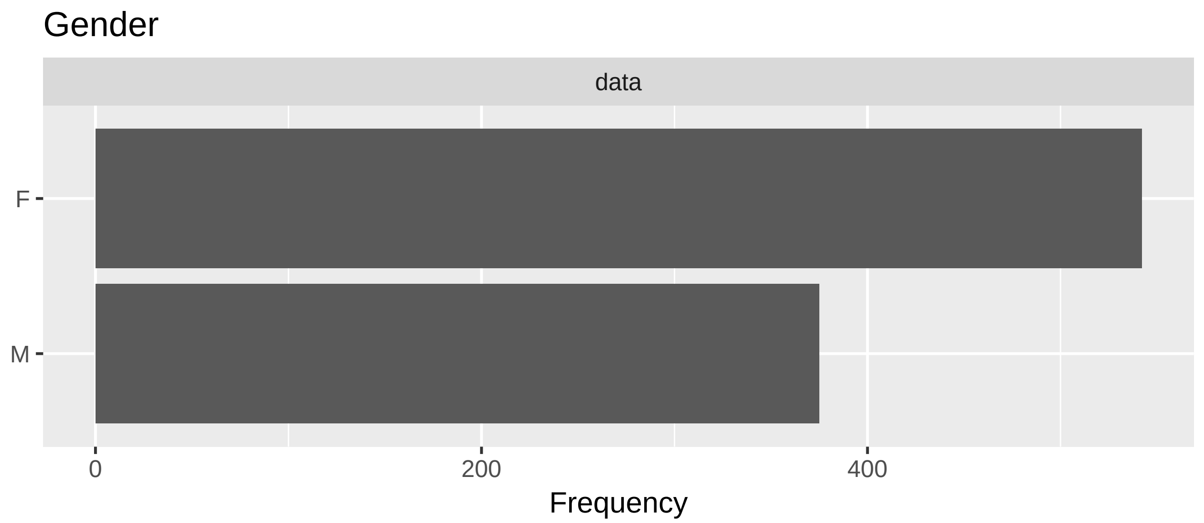


```
data$Gender<- with(data, ifelse(Title %in% "Dr","M",
                               ifelse(Title %in% "Mr","M",
                                      ifelse(Title %in% "Ms","F",
                                             ifelse(Title %in% "Mrs","F","F")))))

data$Gender<-factor(data$Gender)
```

New variable gender with levels M indicating Male and F indicating female

```
plot_bar(data$Gender, title="Gender")
```



The variable Booked.Status is the target variable and it would be ideal to convert it into '1' and '0' before modelling

```
data$Booked.Status<-with(data,ifelse(Booked.Status %in% "YES","1","0"))
data$Booked.Status<-factor(data$Booked.Status)
```

Final check

```
str(data)
```

```
## 'data.frame':    917 obs. of  29 variables:
##  $ Allocated.Time        : Factor w/ 3 levels "Extremely Fast",..: 1 1 1 3 3 1 1 3 1 3 ...
##  $ Web.or.Phone          : Factor w/ 2 levels "PHONE","WEB": 1 1 1 2 2 2 2 2 1 2 ...
##  $ Answered.by.specialist: Factor w/ 2 levels "0","1": 1 1 1 1 1 2 2 1 1 1 ...
##  $ Holiday.Type          : Factor w/ 6 levels "A","B","C","D",..: 1 3 1 1 1 2 1 1 2 1 ...
##  $ Accom.type            : Factor w/ 4 levels "grade1","grade2",..: 1 1 1 1 1 1 2 2 2 1 ...
##  $ Dep.Airport           : Factor w/ 8 levels "Any Airport",..: 4 1 3 7 5 5 4 4 4 7 ...
##  $ Lead.Time             : int  50 14 40 13 74 66 42 39 50 39 ...
##  $ Destination           : Factor w/ 15 levels " AA Resort"," AB",..: 7 2 3 2 3 2 2 3 3 7 ...
##  $ Duration              : num  14 10 14 14 14 14 10 14 13 14 ...
##  $ Adults                : int  6 2 4 2 7 6 2 3 2 7 ...
##  $ Children              : int  2 2 1 1 1 2 0 1 2 2 ...
##  $ Transport.Type        : Factor w/ 3 levels "A","B","None Required": 1 3 1 1 1 1 3 2 2 1 ...
##  $ Answered.Q            : Factor w/ 2 levels "NO","YES": 2 1 1 2 2 2 1 2 2 2 ...
##  $ Notes.Completed       : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Title                 : Factor w/ 5 levels "Dr","Miss","Mr",..: 4 4 4 4 3 4 4 2 4 4 ...
##  $ Enquiry.Comments      : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Booked.Status         : Factor w/ 2 levels "0","1": 2 2 2 1 1 1 1 1 2 1 ...
##  $ EnquiryYear           : Factor w/ 2 levels "2017","2018": 1 1 1 1 1 1 1 1 1 1 ...
##  $ EnquiryMonth          : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ EnquiryDay            : int  1 1 1 1 1 2 3 3 4 4 ...
##  $ EnquiryWeekday        : Factor w/ 7 levels "Friday","Monday",..: 4 4 4 4 4 2 6 6 7 7 ...
##  $ DepYear               : Factor w/ 5 levels "2017","2018",..: 1 1 1 1 2 2 1 1 1 1 ...
##  $ DepMonth              : Factor w/ 12 levels "1","2","3","4",..: 12 4 10 4 6 4 10 10 12 10 ...
##  $ DepDay                : int  19 10 14 8 7 11 22 5 20 7 ...
##  $ DepWeekday            : Factor w/ 7 levels "Friday","Monday",..: 6 2 3 3 5 7 4 5 7 3 ...
##  $ Enquiry.Timecat       : Factor w/ 2 levels "Business_Hour",..: 1 1 1 2 2 2 2 1 1 1 ...
##  $ Enquiry.Time_class    : Factor w/ 3 levels "afternoon","morning",..: 1 2 1 2 2 3 3 3 2 2 ...
##  $ DepartureSeason       : Factor w/ 4 levels "winter","spring",..: 1 2 4 2 3 2 4 4 1 4 ...
##  $ Gender                : Factor w/ 2 levels "F","M": 1 1 1 1 2 1 1 1 1 1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Minor changes (convert Duration to an integer)

```
data$Duration<-as.integer(data$Duration)
```

Save the cleaned data as a csv

```
write.csv(data,file='ReadyforModelling.csv')
```