Exploratory Data Analysis

Chindu

This is a Exploratory Data Analysis report carried out on a sample CRM dataset

The dataset consist of enquiries carried out by people regarding holiday packages over two years. This dataset will be analysed to get better insights that could help improve marketing and business decisions. This is a randomly fabricated dataset just for the purpose of demonstracting the power of EDA.

Loading csv file into R studio

```r
data<-read.csv("ReadyforModelling.csv")
```

Checking if R studio has identified the right structure for each variable

```r
str(data)
```

```
## 'data.frame':    917 obs. of  30 variables:
##  $ X                     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 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: int  0 0 0 0 0 1 1 0 0 0 ...
##  $ 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              : int  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         : int  1 1 1 0 0 0 0 0 1 0 ...
##  $ EnquiryYear           : int  2017 2017 2017 2017 2017 2017 2017 2017 2017 2017 ...
##  $ EnquiryMonth          : int  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               : int  2017 2017 2017 2017 2018 2018 2017 2017 2017 2017 ...
##  $ DepMonth              : int  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 "fall","spring",..: 4 2 1 2 3 2 1 1 4 1 ...
##  $ Gender                : Factor w/ 2 levels "F","M": 1 1 1 1 2 1 1 1 1 1 ...
```

Changing structure of wrongly assigned variables and remove variables unrealated to the analysis

```
data$Answered.by.specialist<- factor(data$Answered.by.specialist)
data$Booked.Status<- factor(data$Booked.Status)
data$EnquiryYear<-factor(data$EnquiryYear)
data$DepYear<-factor(data$DepYear)
data$Children<-factor(data$Children)
data$Adults<-factor(data$Adults)
data$X<-NULL
```

Get a better understanding of numeric/interger variables

```
diagnose_numeric(data)
```

```
## # A tibble: 6 x 10
##   variables      min    Q1  mean median    Q3   max  zero minus outlier
##   <chr>        <int> <dbl> <dbl>  <int> <dbl> <int> <int> <int>   <int>
## 1 Lead.Time        1    29 48.6      47    65   121     0     0       4
## 2 Duration         1    13 13.4      14    14    28     0     0     292
## 3 EnquiryMonth     1     3  5.62      5     9    12     0     0       0
## 4 EnquiryDay       1     8 15.8      16    23    31     0     0       0
## 5 DepMonth         1     5  7.16      8     9    12     0     0       0
## 6 DepDay           1     7 15.1      15    22    31     0     0       0
```

From the diagnosis, it is observed that the variable duration has a high number of outliers and that there is no negative values or zero values in the numeric variables.

Get a better understanding of categorical variables

```
diagnose_category(data)
```

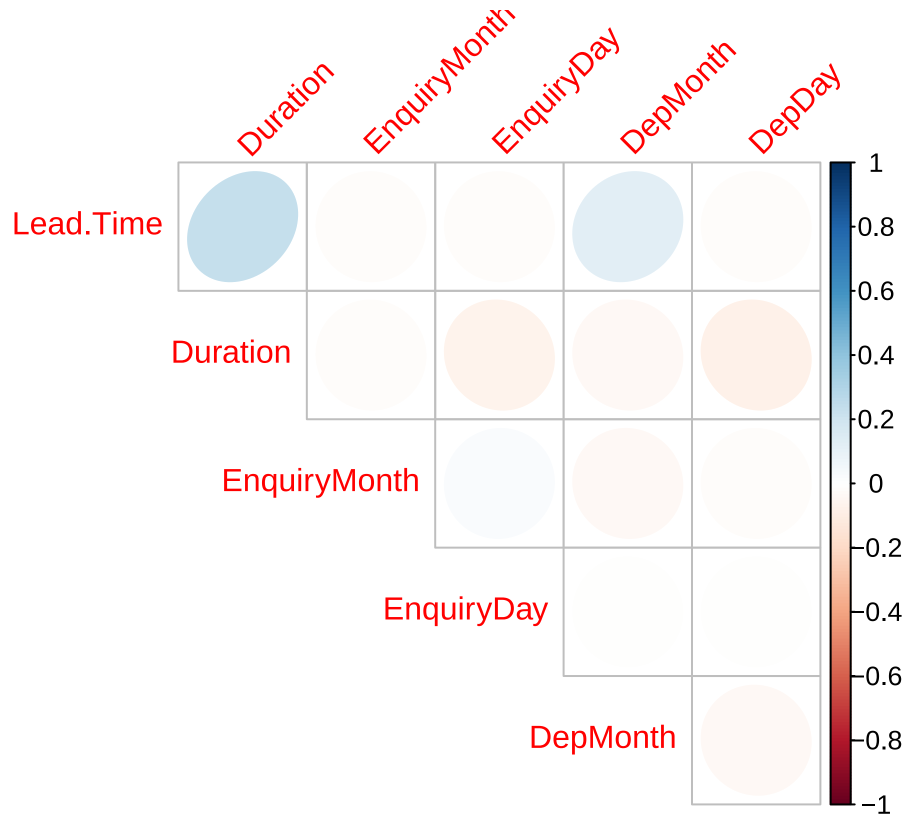
```
## Warning: Factor `variable` contains implicit NA, consider using
## `forcats::fct_explicit_na`
```

```
## # A tibble: 98 x 6
##    variables              levels             N  freq ratio  rank
##    <chr>                  <fct>          <int> <int> <dbl> <int>
##  1 Allocated.Time         Slow             917   553  60.3     1
##  2 Allocated.Time         Extremely Fast   917   242  26.4     2
##  3 Allocated.Time         Fast             917   122  13.3     3
##  4 Web.or.Phone           WEB              917   738  80.5     1
##  5 Web.or.Phone           PHONE            917   179  19.5     2
##  6 Answered.by.specialist 1                917   472  51.5     1
##  7 Answered.by.specialist 0                917   445  48.5     2
##  8 Holiday.Type           A                917   624  68.0     1
##  9 Holiday.Type           B                917   130  14.2     2
## 10 Holiday.Type           E                917   103  11.2     3
## # ... with 88 more rows
```

The diagnosis gives a breakdown of the frequency level and the ratio for each categorical variables. This is useful in understanding rare levels in variables. Example the there are only 9 enquiries each for the Destination LH,LV and SF. Based on information gained from this diagnosis, we could group these three levels together as 'other destinations'.

Checking correlation between numerical variables (fast plot)

```r
plot_correlate(data)
```

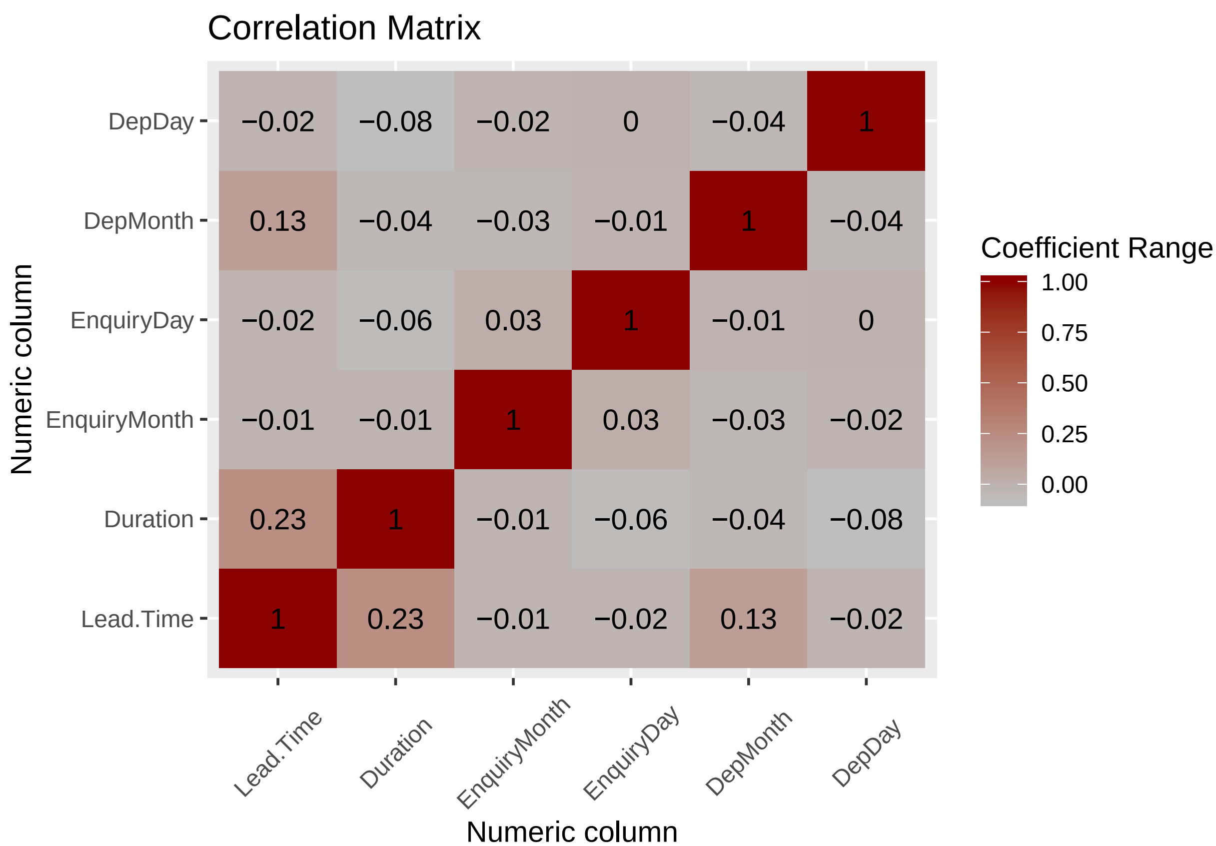


Detailed correlation plot

```r
num.cols<-sapply(data,is.numeric)
data_numcols<-data[,num.cols]
cor(data_numcols)
```

```
##                Lead.Time    Duration EnquiryMonth    EnquiryDay
## Lead.Time     1.00000000  0.23075994  -0.01236704 -0.0198163032
## Duration      0.23075994  1.00000000  -0.01444022 -0.0632792950
## EnquiryMonth -0.01236704 -0.01444022   1.00000000  0.0253698659
## EnquiryDay   -0.01981630 -0.06327930   0.02536987  1.0000000000
## DepMonth      0.12923921 -0.03694669  -0.03419048 -0.0087073505
## DepDay       -0.01792582 -0.07924499  -0.01670854 -0.0005073535
##                 DepMonth        DepDay
## Lead.Time     0.12923921 -0.0179258152
## Duration     -0.03694669 -0.0792449886
## EnquiryMonth -0.03419048 -0.0167085374
## EnquiryDay   -0.00870735 -0.0005073535
## DepMonth      1.00000000 -0.0366504502
## DepDay       -0.03665045  1.0000000000
```

```
melted_corr<-melt(cor(data_numcols))
ggplot(data=melted_corr,aes(x=Var1,y=Var2,fill=value))+
  geom_tile()+
  scale_fill_gradient(low="grey",high="darkred")+
  geom_text(aes(x=Var1,y=Var2,label=round(value,2)),size=4)+
  labs(title="Correlation Matrix",x="Numeric column",y="Numeric column",
       fill="Coefficient Range")+
  theme(axis.text.x=element_text(angle=45, vjust=0.5))
```



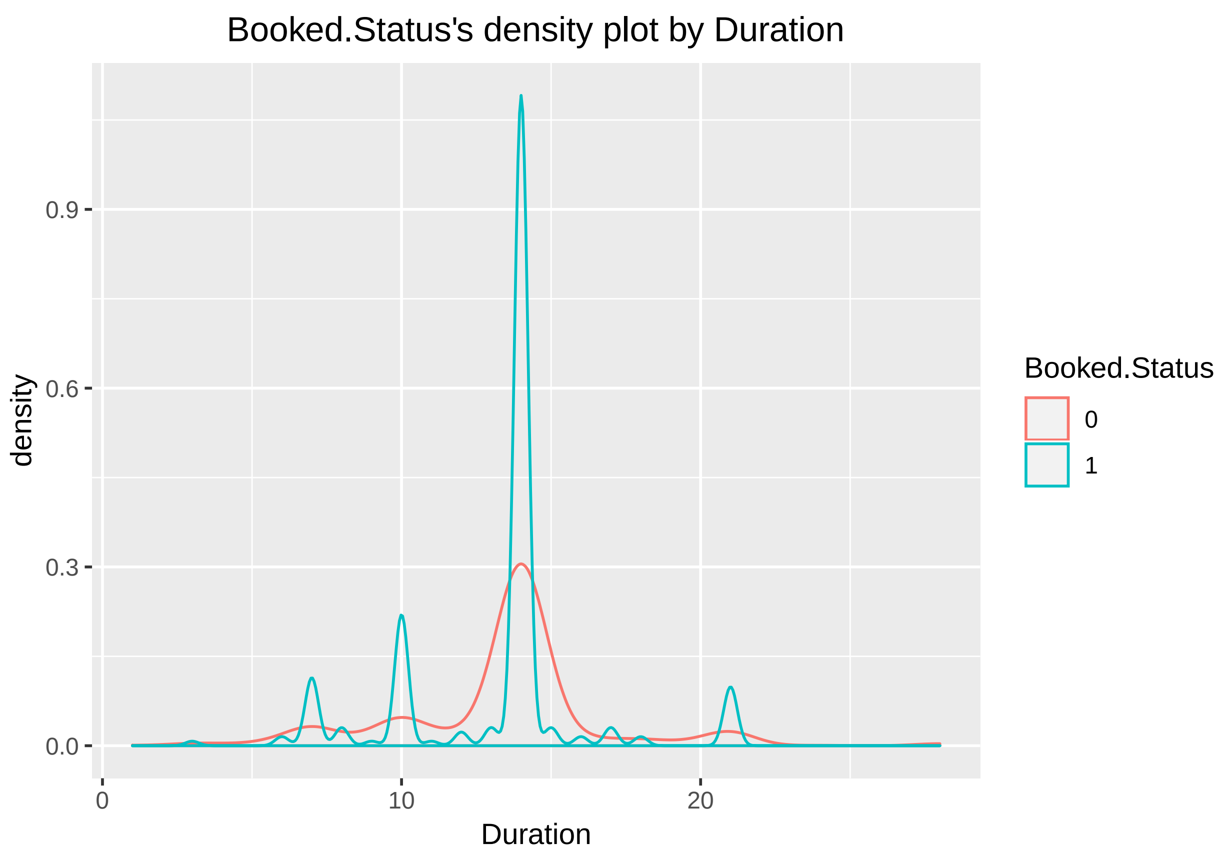
From the correlation plot it is understood that there is little relation between the numeric variable. The strongest relationship is between duration and lead time, but is a rather weak relation.

Exploring relation between target varaible (Booked.Status) and a numeric variable

```
categ<-target_by(data,Booked.Status)
cat_num<-relate(categ,Duration)
```

Relationship between booked.status and duration is represented using a desity plot

```
plot(cat_num)
```



Exploring relation between target variable(BookedStatus) and a categorical variable

```
cat_cat<-relate(categ,Allocated.Time)
cat_cat
```

```
##              Allocated.Time
## Booked.Status Extremely Fast Fast Slow
##             0            100   35  553
##             1            142   87    0
```

```
plot(cat_cat) #mosaics plot
```

By understanding the relationship it is clear that if the Allocated.Time is slow the chances of booking is significantly lowered.

Checking for skewness in numeric variables (If skewness value lies above +1 or below -1, data is highly skewed. If it lies between +0.5 to -0.5, it is moderately skewed. If the value is 0, then the data is symmetric)

```
data %>%
  describe() %>%
  select(variable, skewness) %>%
  filter(!is.na(skewness)) %>%
  arrange(desc(abs(skewness)))
```

```
## # A tibble: 6 x 2
##   variable     skewness
##   <chr>           <dbl>
## 1 Lead.Time      0.420
## 2 DepMonth      -0.379
## 3 EnquiryMonth   0.179
## 4 Duration       0.141
## 5 DepDay         0.0755
## 6 EnquiryDay     0.0384
```

Lead.Time is highly skewed. To reduce the skewness and to achive a distribution that is close to a normal distribution, a sqrt transformation is used.

```
data$sqrt_lead.time<-sqrt(data$Lead.Time)

data %>%
  describe() %>%
  select(variable, skewness) %>%
  filter(!is.na(skewness)) %>%
  arrange(desc(abs(skewness)))
```

```
## # A tibble: 7 x 2
##   variable       skewness
##   <chr>             <dbl>
## 1 Lead.Time        0.420
## 2 DepMonth        -0.379
## 3 sqrt_lead.time  -0.282
## 4 EnquiryMonth     0.179
## 5 Duration         0.141
## 6 DepDay           0.0755
## 7 EnquiryDay       0.0384
```

The skewness for Lead.Time is now reduced.

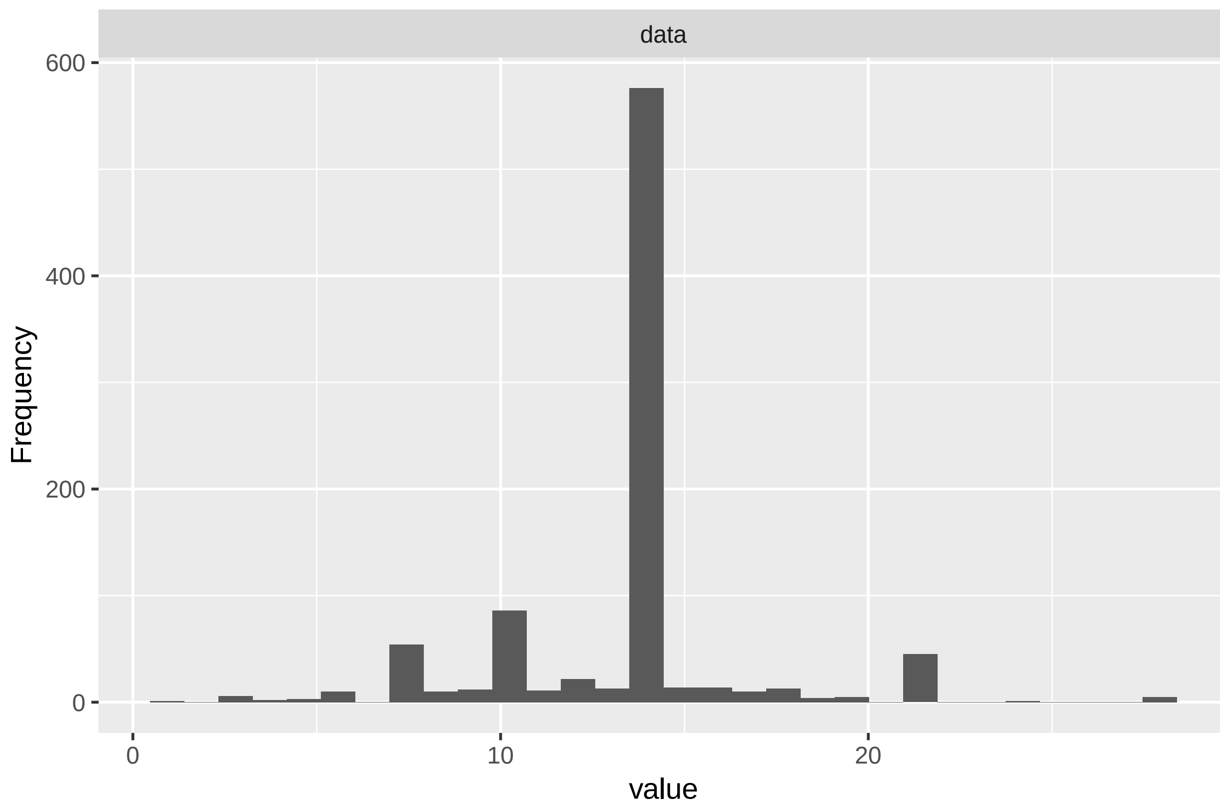
Diagnose anomalies of all numeric variables of data

```r
diagnose_outlier(data)
```

```
##        variables outliers_cnt outliers_ratio outliers_mean with_mean
## 1      Lead.Time            4      0.4362050     120.25000 48.647764
## 2       Duration          292     31.8429662      12.19863 13.379498
## 3   EnquiryMonth            0      0.0000000           NaN  5.622683
## 4     EnquiryDay            0      0.0000000           NaN 15.780807
## 5       DepMonth            0      0.0000000           NaN  7.157034
## 6         DepDay            0      0.0000000           NaN 15.140676
## 7 sqrt_lead.time            2      0.2181025       1.00000  6.706054
##   without_mean
## 1    48.334064
## 2    13.931200
## 3     5.622683
## 4    15.780807
## 5     7.157034
## 6    15.140676
## 7     6.718526
```

The variable duration has approximately 32% observations identified as outliers

```r
plot_histogram(data$Duration)
```



From the plot it is observed that the high skewness is due to majority of enquiries are for 7,10,14 or 21 days. Tabulate the values to get a better understanding.

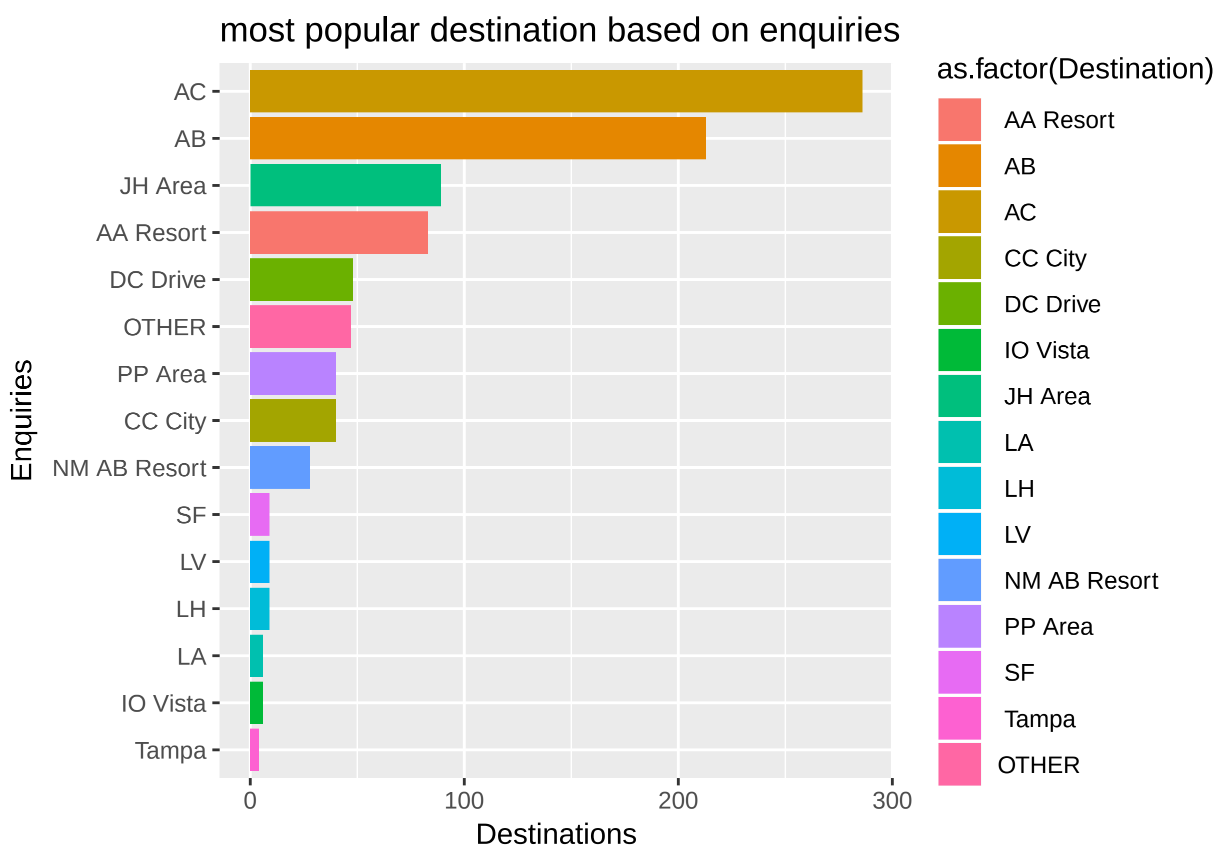
```
table(data$Duration)
```

```
##
##   1   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19
##   1   6   2   3  10  54  10  12  86  11  22  13 576  14  14  10  13   4
##  20  21  24  28
##   5  45   1   5
```

Answering questions using data visualisation techniques

Desination by popularity and what is the total enquiries for each destination?

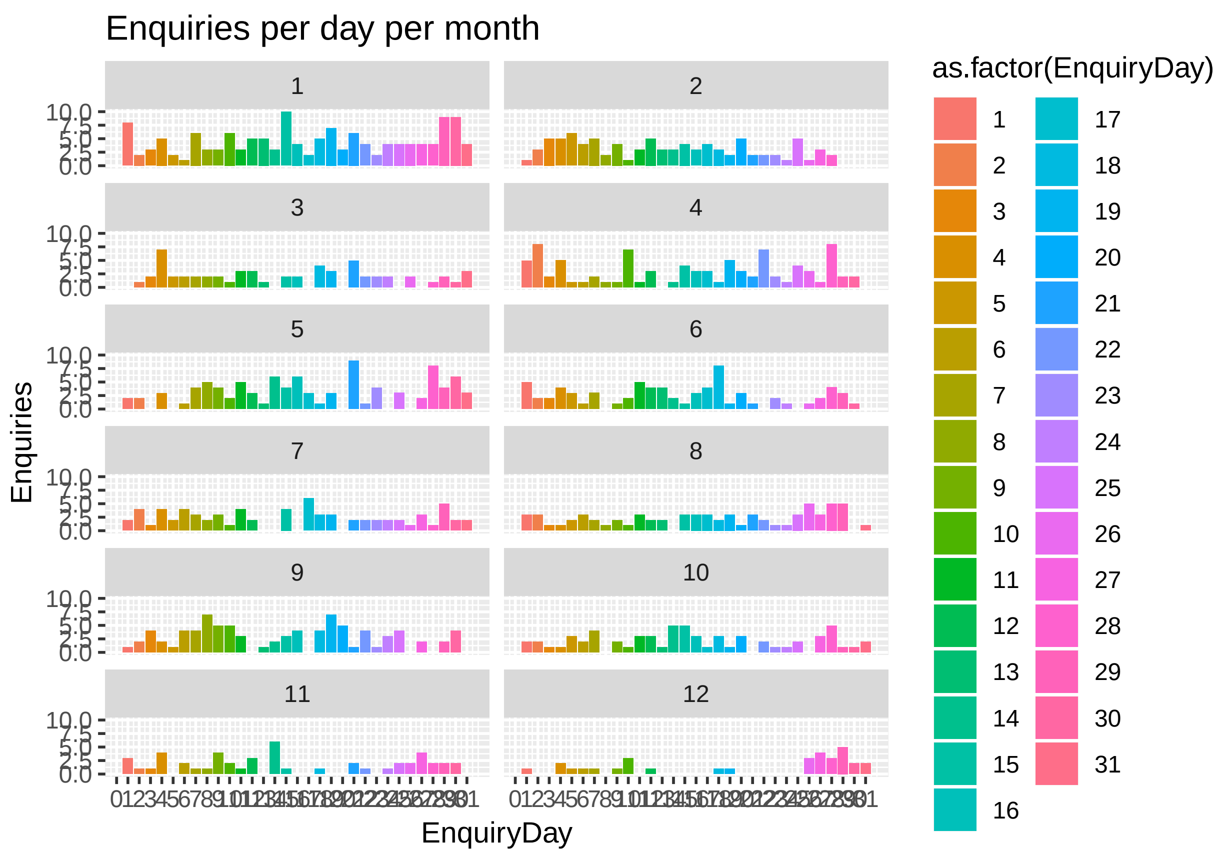
```
pop_destination<- data %>% group_by(Destination) %>% count(Destination) %>%ungroup()

ggplot(data=pop_destination,aes(x=reorder(as.factor(Destination),n),
       y=n,fill=as.factor(Destination)))+geom_bar(stat="identity")+coord_flip()+
labs(title= "most popular destination based on enquiries", x="Enquiries",y="Destinations")
```



From the plot we know that the two most popular destinations are AC and AB. The least popular destinations are LA and Fort Lauderdale.

What are the day and month wise total enquiries?

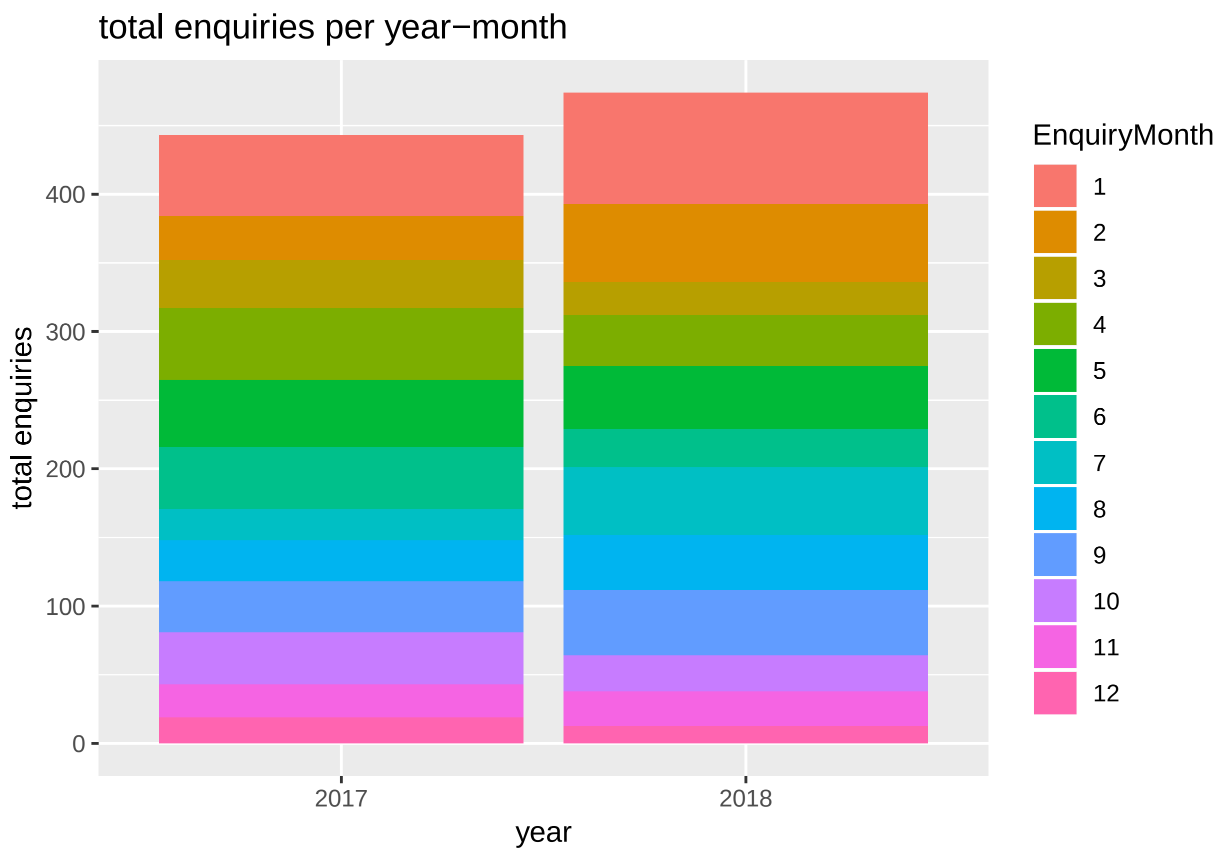
```
day_month_sale<-data%>%group_by(EnquiryMonth,EnquiryDay) %>%
  count(Destination)%>%arrange(EnquiryMonth,EnquiryDay) %>% ungroup()

ggplot(data=day_month_sale, aes(x=EnquiryDay,y=n,fill=as.factor(EnquiryDay)))+
  geom_bar(stat="identity")+scale_x_continuous(breaks=seq(min(0),max(31),by=1))+
  facet_wrap(~EnquiryMonth,ncol=2)+
labs(title= "Enquiries per day per month", x="EnquiryDay",y="Enquiries")
```



By understanding the plot, the company can allocate more agents to attend enquiries on specific days of the months where the number of enquiries are high. For example in January(1) more agents are required in the begining of the month, middle and towards the end of the month. Assigning more agents during these time would improve the Allocation.time and could lead to increase in booking.

Total enquiries by year and month

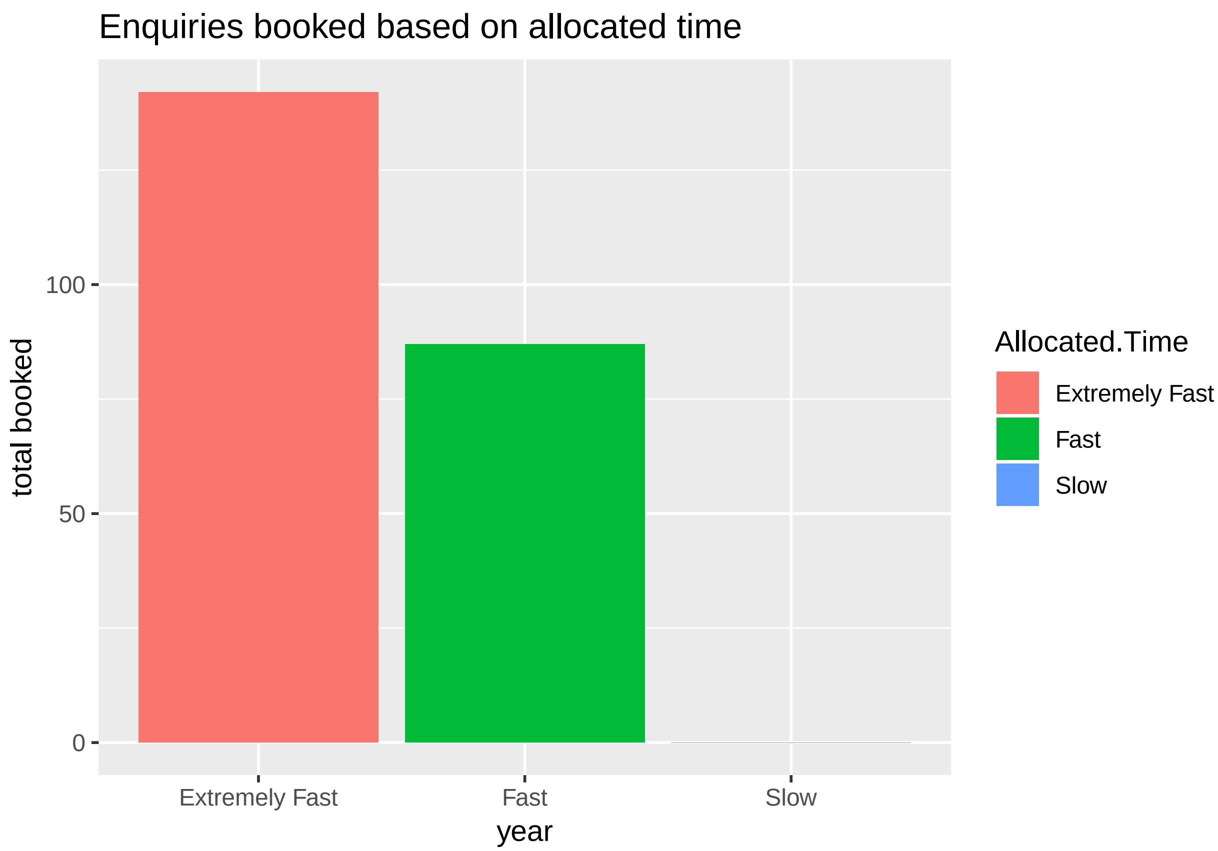
```
year_month<- data%>%group_by(EnquiryYear,EnquiryMonth) %>%
  count(Destination)%>%arrange(EnquiryYear)%>%ungroup()
ggplot(data=year_month,aes(x=EnquiryYear,y=n,fill=as.factor(EnquiryMonth)))+
  geom_bar(stat="identity")+labs(title="total enquiries per year-month",
                                 x="year",y="total enquiries",fill="EnquiryMonth")
```



By analysing this plot we are able to understand that generally the most number of enquiries comes in during the first few months of the year. In December and march the number of enquiries are generally lower and would be an ideal time for employees to clear their holiday entitlement.

How many enquiries were booked based on Allocated.Time?

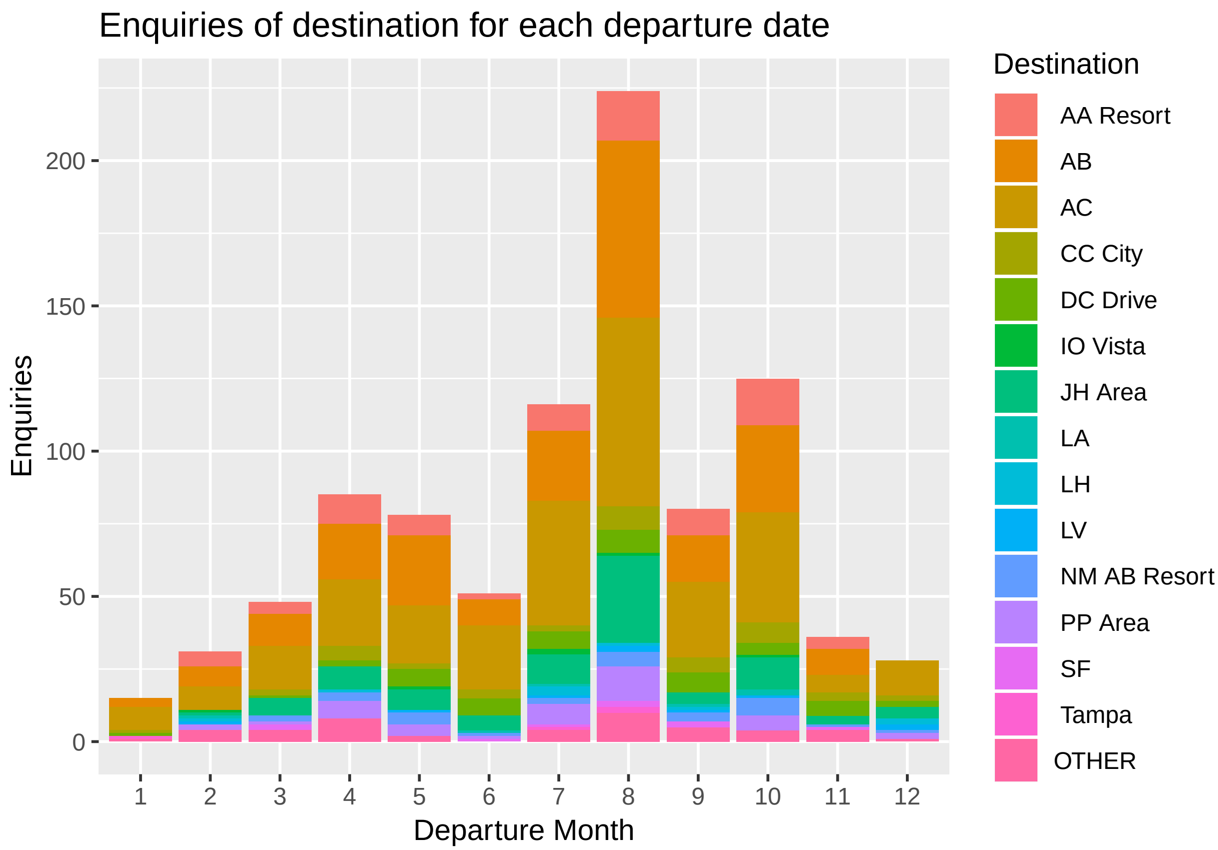
```
data$Booked.Status<-as.integer(data$Booked.Status)
data$Booked.Status<-ifelse(data$Booked.Status %in% 1,0,1)
booked_Allocated<-data%>%group_by(Allocated.Time)%>% summarise(booked=sum(Booked.Status))%>%
  arrange(Allocated.Time)%>%ungroup()
ggplot(data=booked_Allocated,aes(x=Allocated.Time,y=booked,fill=as.factor(Allocated.Time)))+
  geom_bar(stat="identity")+
  labs(title="Enquiries booked based on allocated time",
       x="year",y="total booked",fill="Allocated.Time")
```



This plot clearly shows that allocated time plays a significant part in an enquiry being booked. If an enquiry is attended to with a allocated time of 'slow' the potential customer will likely to seek other companies for their holiday packages.

Which destinations are popular based on departure months?

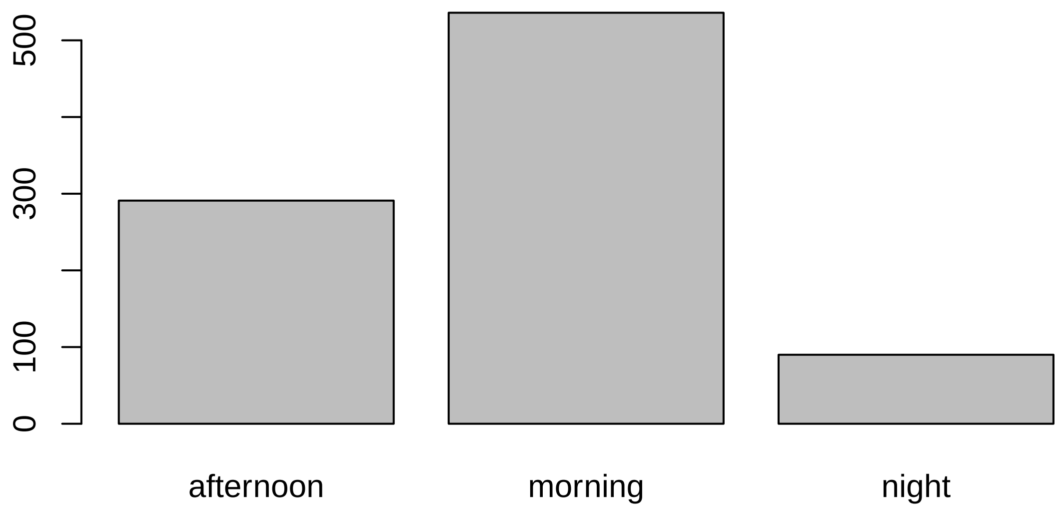
```
ggplot(data,aes(x=factor(data$DepMonth),fill=Destination))+geom_bar()+
  labs(title="Enquiries of destination for each departure date",
       x="Departure Month",y="Enquiries",fill="Destination")
```



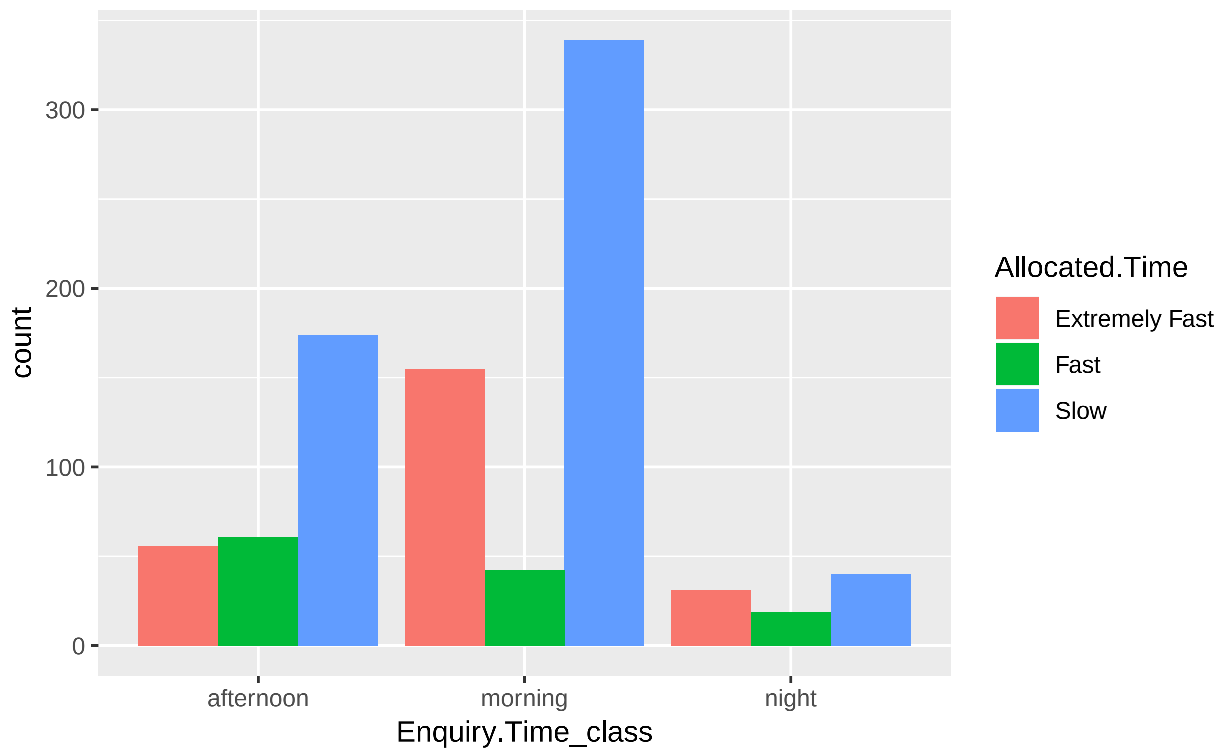
As we say earlier AB and AC are the most popular destination, this plots gives a breakdown on when each destination is more popular. This plot can help the marketing team to plan promotion pakages for the various months to improve business.

Which time of the day is the most enquiries coming in and Which period of the day is the Allocated.Time the worst?

```
plot(data$Enquiry.Time_class)
```




```
ggplot(data,aes(x=Enquiry.Time_class,fill=Allocated.Time)) + geom_bar(position="dodge")
```



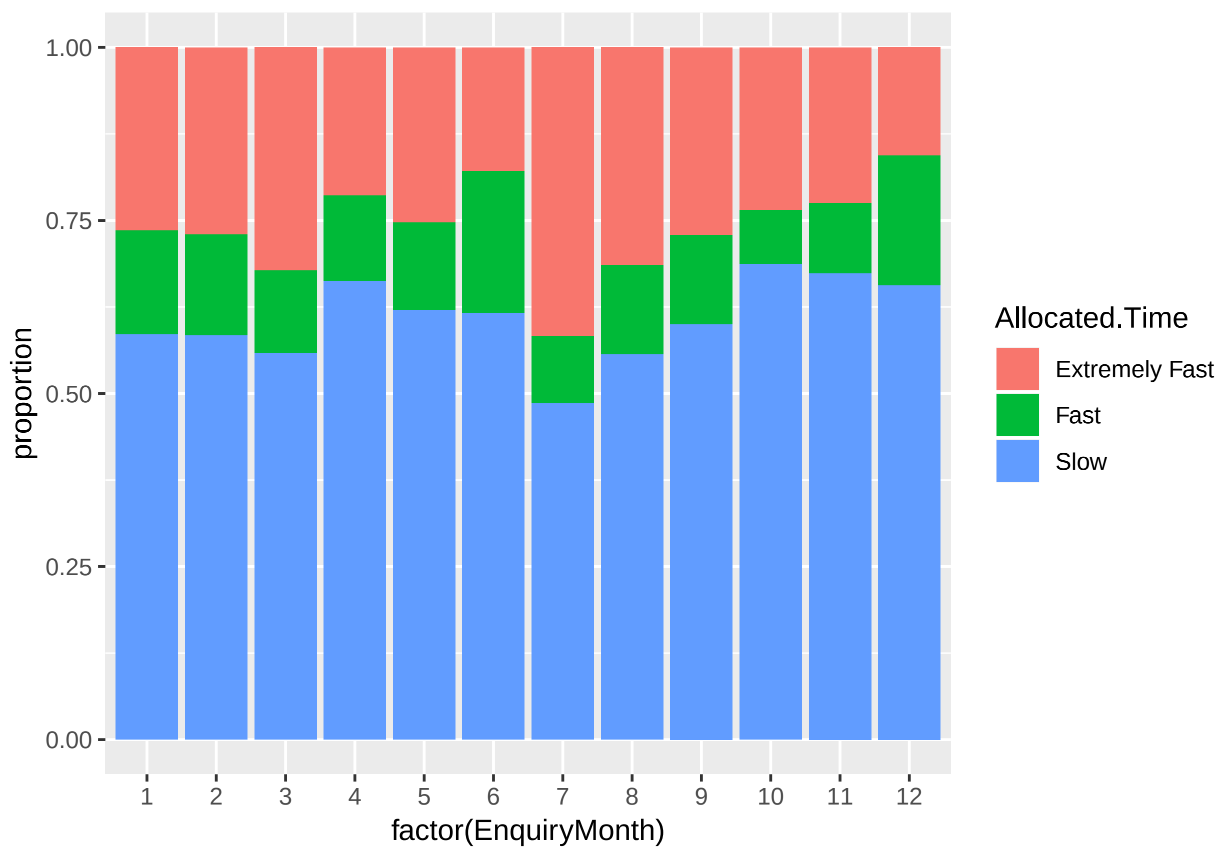
This plot shows that the majority of the enquiries are received in the morning and that there are not enough agents available to deal with all the enquiries at an optimal speed, resulting in a huge number of enquiries with an allocated.time of ‘slow’. A strategy should be devised to improve this situation.

what is the Proportion of agent allocation speed for morning, afternoon and night?

```
tab_count<-table(data$EnquiryMonth,data$Allocated.Time)
prop.table(tab_count,1)
```

```
##
##      Extremely Fast       Fast       Slow
##   1      0.26428571 0.15000000 0.58571429
##   2      0.26966292 0.14606742 0.58426966
##   3      0.32203390 0.11864407 0.55932203
##   4      0.21348315 0.12359551 0.66292135
##   5      0.25263158 0.12631579 0.62105263
##   6      0.17808219 0.20547945 0.61643836
##   7      0.41666667 0.09722222 0.48611111
##   8      0.31428571 0.12857143 0.55714286
##   9      0.27058824 0.12941176 0.60000000
##   10     0.23437500 0.07812500 0.68750000
##   11     0.22448980 0.10204082 0.67346939
##   12     0.15625000 0.18750000 0.65625000
```

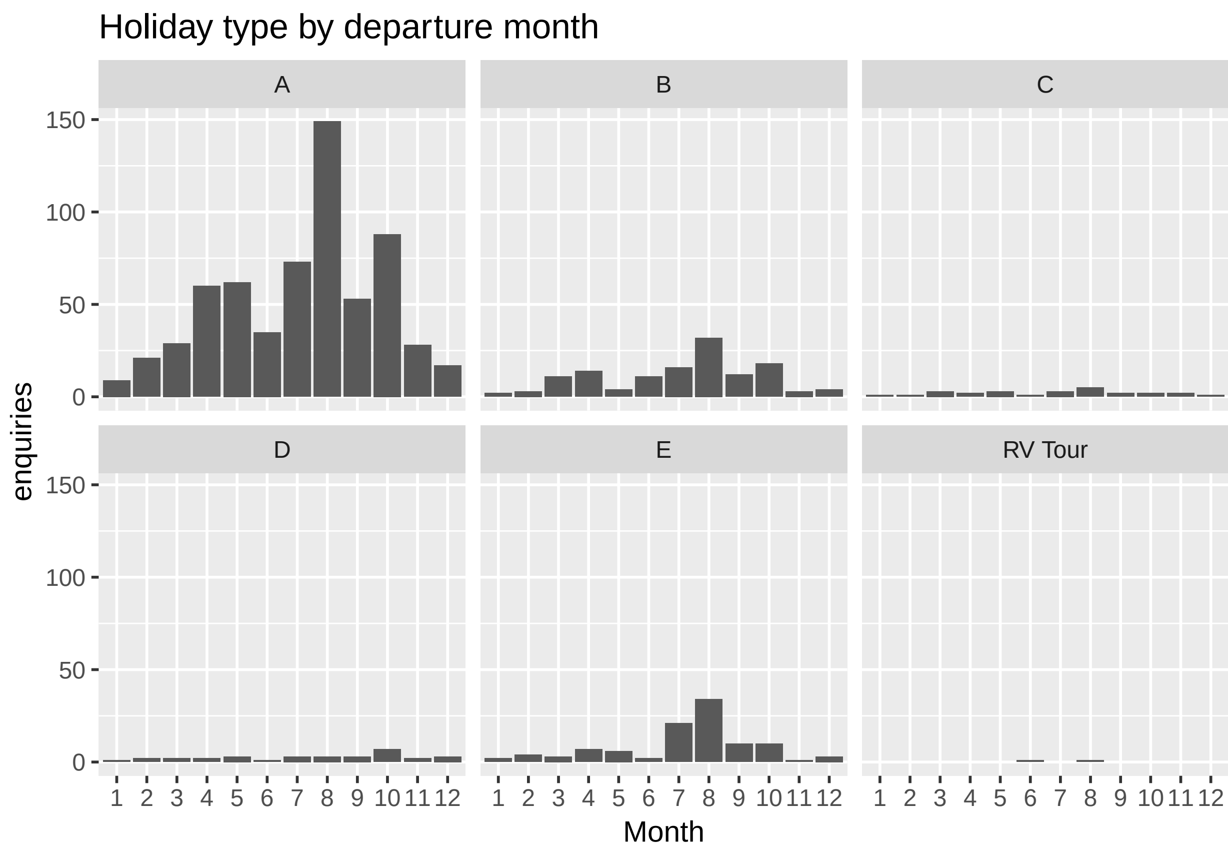
```
ggplot(data,aes(x=factor(EnquiryMonth),fill=Allocated.Time)) +
  geom_bar(position="fill") + ylab("proportion")
```



This plot can be used to plan holiday entitlement to employees. Holiday entitlement should be reduced for months were Allocated.time is high. The major problem seems to be occuring in June and Decmber.

Which months are popular for each holiday type?

```
ggplot(data,aes(x=factor(DepMonth))) + geom_bar() + facet_wrap(~Holiday.Type) +
  labs(title="Holiday type by departure month",x="Month",y="enquiries")
```



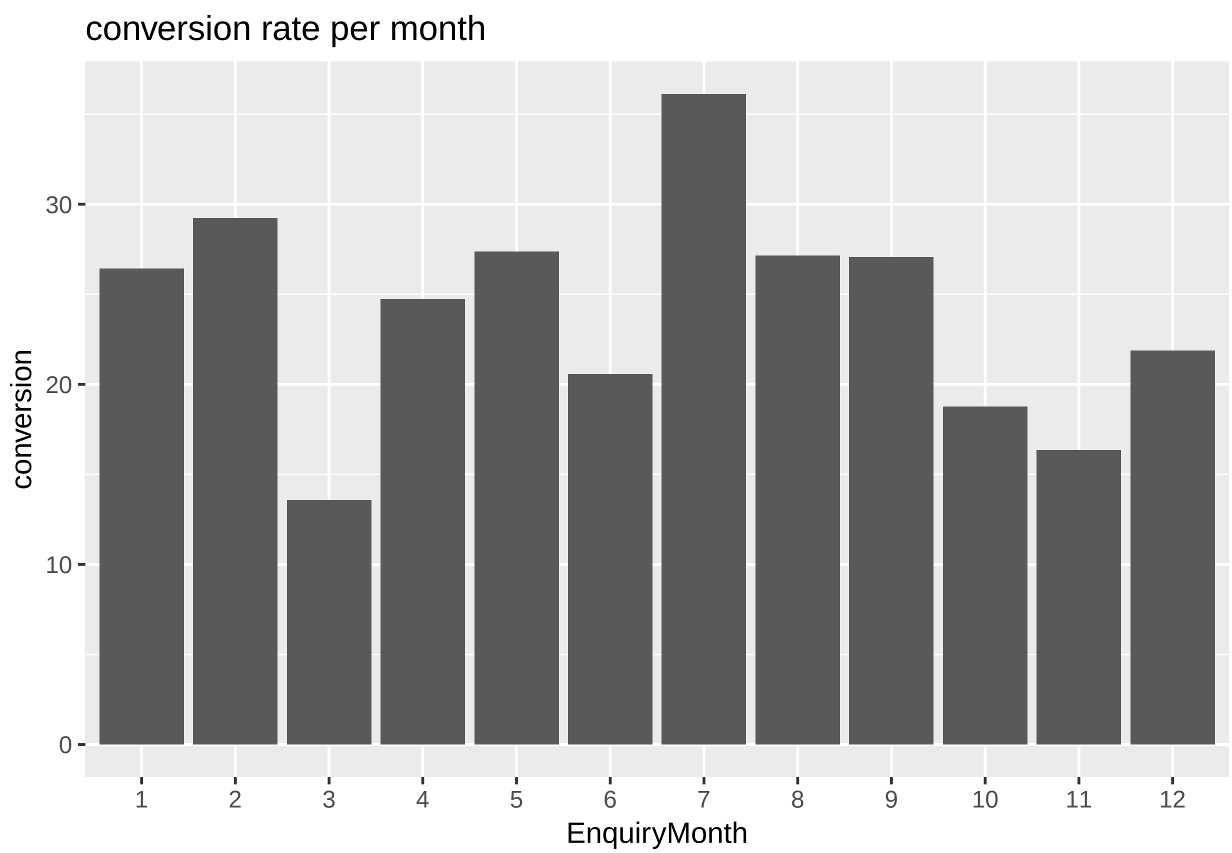
This plot can be used to introduce promotional holiday packages for the various months.

What is the conversion rate per month?

```r
data$Booked<-as.integer(data$Booked.Status)
summarization <- sqldf("select EnquiryMonth, count(EnquiryMonth) as enquiries,
                       sum(Booked) as totalbooked from data group by EnquiryMonth")
summarization$totalbooked<- as.numeric(summarization$totalbooked)
summarization$enquiries<- as.numeric(summarization$enquiries)
conversionrate <- sqldf("select *,
                        (totalbooked/enquiries)*100 as conversion from summarization")
data.frame(conversionrate)
```

```
##    EnquiryMonth enquiries totalbooked conversion
## 1             1       140          37   26.42857
## 2             2        89          26   29.21348
## 3             3        59           8   13.55932
## 4             4        89          22   24.71910
## 5             5        95          26   27.36842
## 6             6        73          15   20.54795
## 7             7        72          26   36.11111
## 8             8        70          19   27.14286
## 9             9        85          23   27.05882
## 10           10        64          12   18.75000
## 11           11        49           8   16.32653
## 12           12        32           7   21.87500
```

```r
ggplot(conversionrate,aes(x=factor(EnquiryMonth),y=conversion))+
  geom_bar(stat="identity")+labs(title="conversion rate per month",x="EnquiryMonth")
```



Conversion rate relates to the profit earned by the company each month. From the plot we can determine which months the company is making the most profit.