DS_Project

2023-03-20

```
library(tidyverse)
library(modelr)
library(lubridate)
library(dplyr)
library(glmnet)
library(caret)
library(pROC)

economy_data = read_csv("Flights Price Prediction Dataset\\economy.csv", show_col_types = FALSE)
business_data = read_csv("Flights Price Prediction Dataset\\business.csv", show_col_types = FALSE)
flight = read_csv("Flights Price Prediction Dataset\\Clean_Dataset.csv", show_col_types = FALSE)
economy_data$time_taken = lubridate::hm(economy_data$time_taken)
business_data$time_taken = lubridate::hm(business_data$time_taken)
head(economy_data)
```

```
## # A tibble: 6 x 11
##   date   airline ch_code num_code dep_time from  time_taken stop  arr_time to
##   <chr>  <chr>   <chr>      <dbl> <time>   <chr> <Period>   <chr> <time>   <chr>
## 1 11-02~ SpiceJ~ SG          8709 18:55    Delhi 2H 10M 0S  non-~ 21:05    Mumb~
## 2 11-02~ SpiceJ~ SG          8157 06:20    Delhi 2H 20M 0S  non-~ 08:40    Mumb~
## 3 11-02~ AirAsia I5           764 04:25    Delhi 2H 10M 0S  non-~ 06:35    Mumb~
## 4 11-02~ Vistara UK           995 10:20    Delhi 2H 15M 0S  non-~ 12:35    Mumb~
## 5 11-02~ Vistara UK           963 08:50    Delhi 2H 20M 0S  non-~ 11:10    Mumb~
## 6 11-02~ Vistara UK           945 11:40    Delhi 2H 20M 0S  non-~ 14:00    Mumb~
## # i 1 more variable: price <dbl>
```

```
head(business_data)
```

```
## # A tibble: 6 x 11
##   date   airline ch_code num_code dep_time from  time_taken stop  arr_time to
##   <chr>  <chr>   <chr>      <dbl> <time>   <chr> <Period>   <chr> <time>   <chr>
## 1 11-02~ Air In~ AI           868 18:00    Delhi 2H 0M 0S   "non~ 20:00    Mumb~
## 2 11-02~ Air In~ AI           624 19:00    Delhi 2H 15M 0S  "non~ 21:15    Mumb~
## 3 11-02~ Air In~ AI           531 20:00    Delhi 24H 45M 0S "1-s~ 20:45    Mumb~
## 4 11-02~ Air In~ AI           839 21:25    Delhi 26H 30M 0S "1-s~ 23:55    Mumb~
## 5 11-02~ Air In~ AI           544 17:15    Delhi 6H 40M 0S  "1-s~ 23:55    Mumb~
## 6 11-02~ Vistara UK           985 19:50    Delhi 2H 10M 0S  "non~ 22:00    Mumb~
## # i 1 more variable: price <dbl>
```

```
flight["stops"][flight["stops"]== "zero"] <- '0'
flight["stops"][flight["stops"]== "one"] <- '1'
flight["stops"][flight["stops"]== "two_or_more"] <- '2'
flight$stops = as.numeric(flight$stops)
typeof(flight$stops)
```

```
## [1] "double"
```

```
head(flight)
```

```
## # A tibble: 6 x 13
##    ...1 airline  flight  source_city departure_time stops arrival_time
##   <dbl> <chr>    <chr>   <chr>       <chr>          <dbl> <chr>
## 1     0 SpiceJet SG-8709 Delhi       Evening            0 Night
## 2     1 SpiceJet SG-8157 Delhi       Early_Morning      0 Morning
## 3     2 AirAsia  I5-764  Delhi       Early_Morning      0 Early_Morning
## 4     3 Vistara  UK-995  Delhi       Morning            0 Afternoon
## 5     4 Vistara  UK-963  Delhi       Morning            0 Morning
## 6     5 Vistara  UK-945  Delhi       Morning            0 Afternoon
## # i 6 more variables: destination_city <chr>, class <chr>, duration <dbl>,
## #   days_left <dbl>, price <dbl>, date <chr>
```

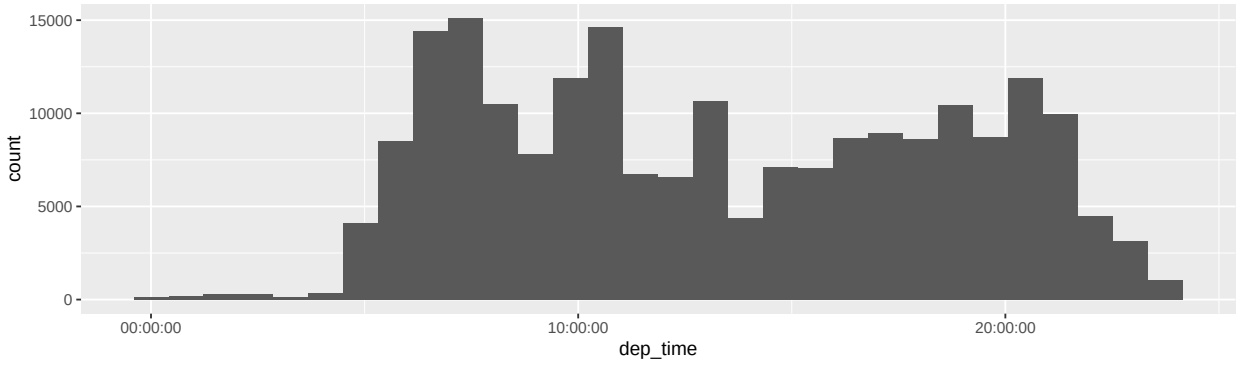
```
flight$sector = paste(flight$source_city, flight$destination_city)
flight$date <- as.Date(flight$date)
flight$weekday <- strftime(flight$date,"%A")
head(flight)
```

```
## # A tibble: 6 x 15
##    ...1 airline  flight  source_city departure_time stops arrival_time
##   <dbl> <chr>    <chr>   <chr>       <chr>          <dbl> <chr>
## 1     0 SpiceJet SG-8709 Delhi       Evening            0 Night
## 2     1 SpiceJet SG-8157 Delhi       Early_Morning      0 Morning
## 3     2 AirAsia  I5-764  Delhi       Early_Morning      0 Early_Morning
## 4     3 Vistara  UK-995  Delhi       Morning            0 Afternoon
## 5     4 Vistara  UK-963  Delhi       Morning            0 Morning
## 6     5 Vistara  UK-945  Delhi       Morning            0 Afternoon
## # i 8 more variables: destination_city <chr>, class <chr>, duration <dbl>,
## #   days_left <dbl>, price <dbl>, date <date>, sector <chr>, weekday <chr>
```

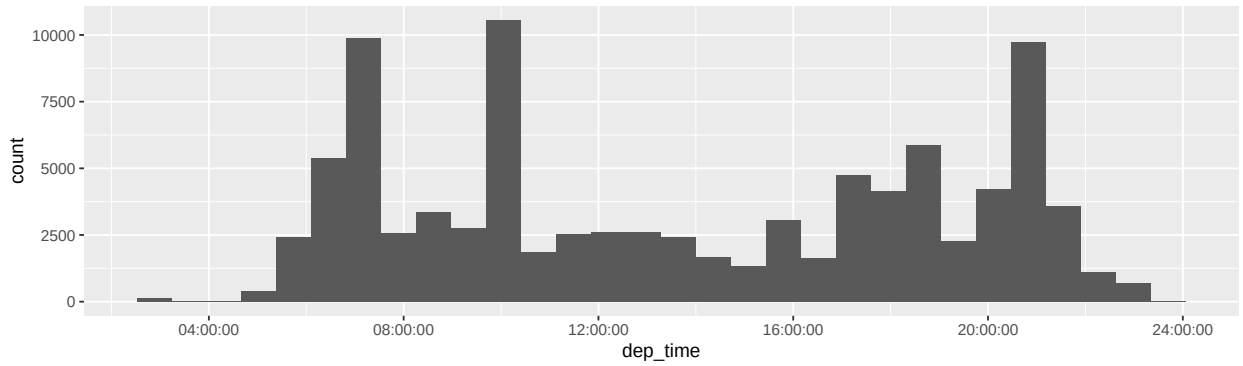
##Distribution of data with respect to departure time

```
#distribution with respect to departure time

ggplot (data = economy_data) +
  geom_histogram(mapping = aes(x=dep_time))
```

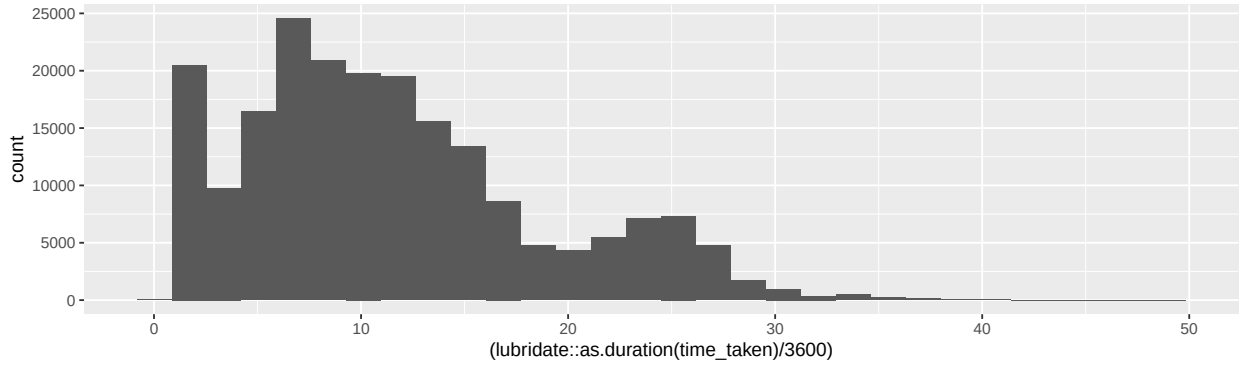


```
ggplot (data = business_data) +
  geom_histogram(mapping = aes(x=dep_time))
```

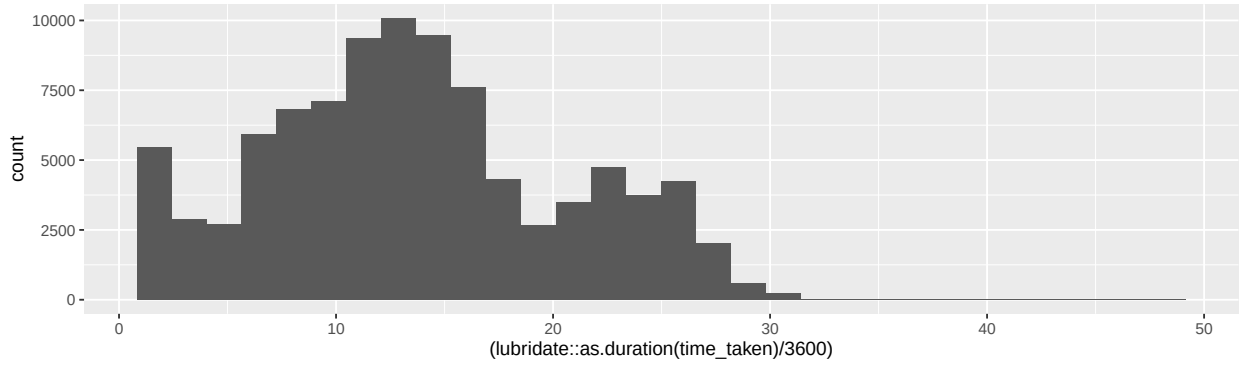


##Distribution of data with respect to duration

```
ggplot (data = economy_data) +
  geom_histogram(mapping = aes(x=(lubridate::as.duration(time_taken)/3600)))
```

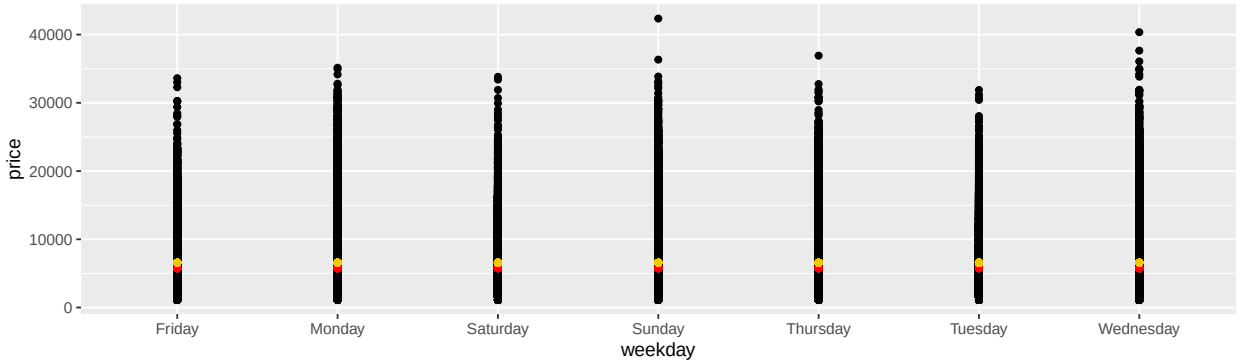



```
ggplot (data = business_data) +
  geom_histogram(mapping = aes(x=(lubridate::as.duration(time_taken)/3600)))
```

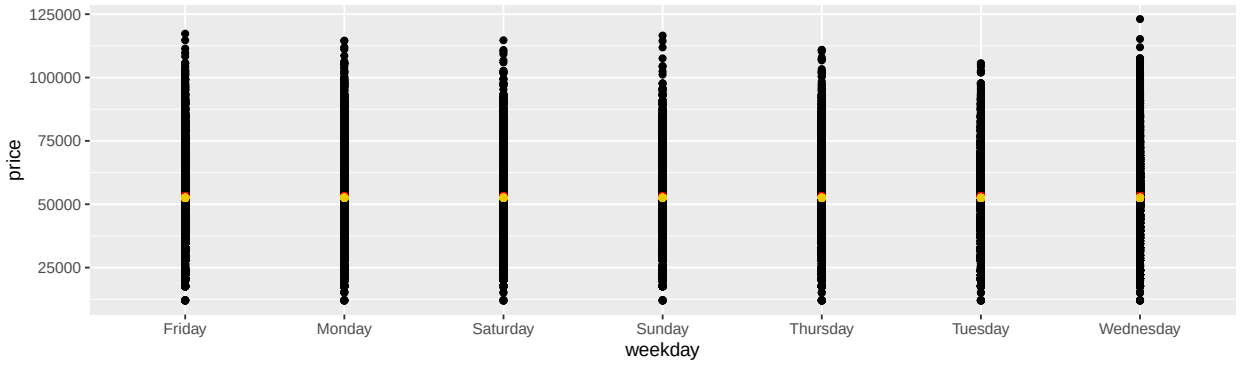


##Price distribution based on the day of the week

```
  #Converting date to weekday
economy_data$date <- as.Date(economy_data$date)
economy_data$weekday <- strftime(economy_data$date,"%A")
business_data$date <- as.Date(business_data$date)
business_data$weekday <- strftime(business_data$date,"%A")
  #Plotting price vs day of economy
weekday_graph_economy <- ggplot (data = economy_data) +
  geom_point(mapping = aes (x = weekday, y = price))+
  geom_point(mapping = aes (x = weekday, y = median(price)),color="red")+
  geom_point(mapping = aes (x = weekday, y = mean(price)),color="gold2")
(weekday_graph_economy)
```

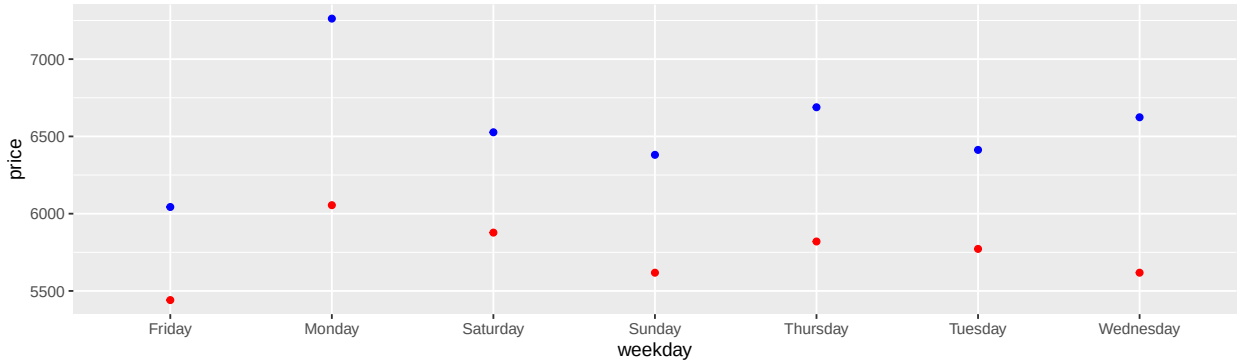



```
  #Plotting price vs day of business
weekday_graph_business <- ggplot (data = business_data) +
  geom_point(mapping = aes (x = weekday, y = price))+
  geom_point(mapping = aes (x = weekday, y = median(price)),color="red")+
  geom_point(mapping = aes (x = weekday, y = mean(price)),color="gold2")
(weekday_graph_business)
```

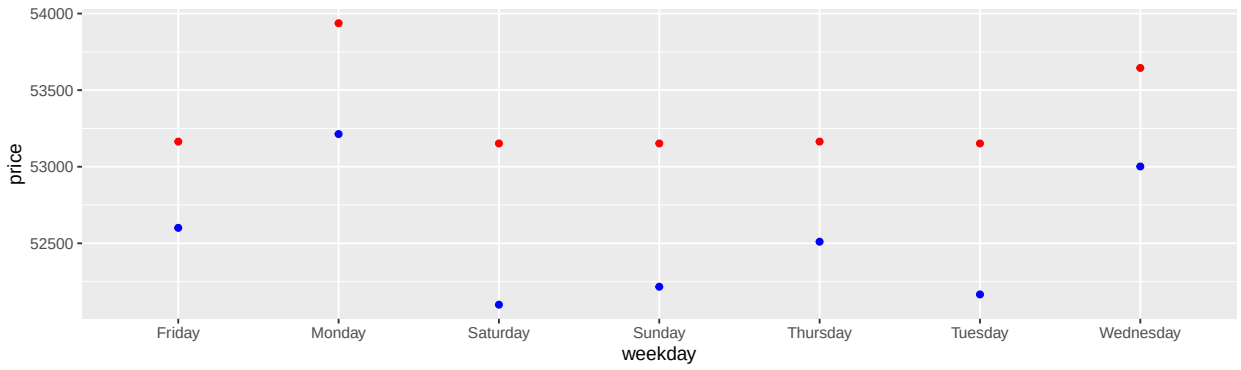



```
  #Grouping dataset by weekday
eco_grp_weekday = economy_data %>% group_by(weekday)  %>%
  summarise(mean_price = mean(price),
            median_price = median(price),
            .groups = 'drop')
bus_grp_weekday = business_data %>% group_by(weekday)  %>%
  summarise(mean_price = mean(price),
            median_price = median(price),
```

```
            .groups = 'drop')
    #Mean and median price in economy by weekday
grp_economy <- ggplot (data = eco_grp_weekday) +
  geom_point(mapping = aes (x = weekday, y = mean_price),color="blue")+
  geom_point(mapping = aes (x = weekday, y = median_price),color="red")+
  ylab("price")
(grp_economy)
```

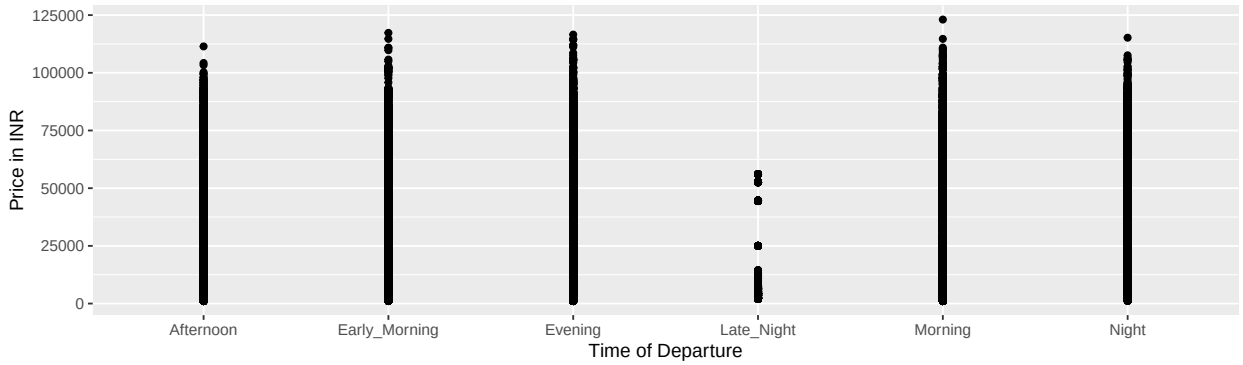



```
    #Mean and median price in business by weekday
grp_business <- ggplot (data = bus_grp_weekday) +
  geom_point(mapping = aes (x = weekday, y = mean_price),color="blue")+
  geom_point(mapping = aes (x = weekday, y = median_price),color="red")+
  ylab("price")
(grp_business)
```



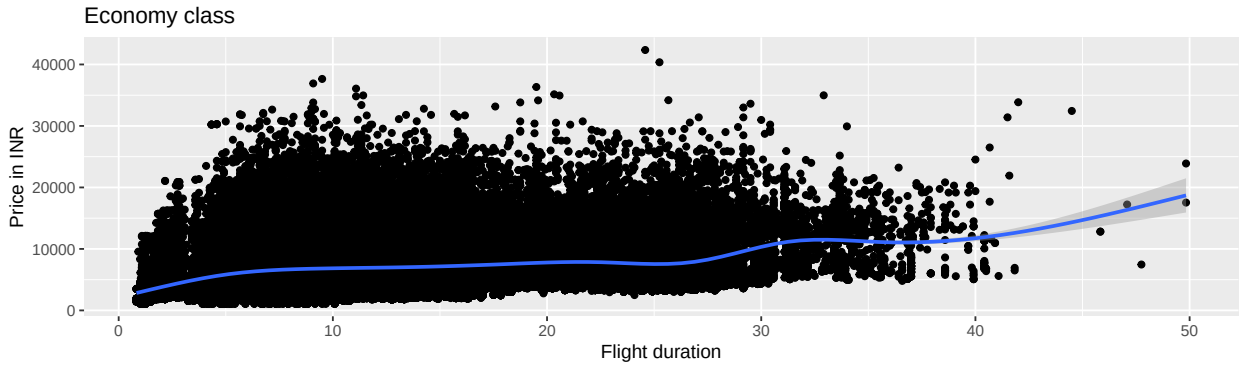
##Price distribution on the basis of time in the day (currently merged business and economy)

```
ggplot(data = flight)+
  geom_point(mapping = aes(departure_time, price))+
  labs(y= "Price in INR", x = "Time of Departure")
```

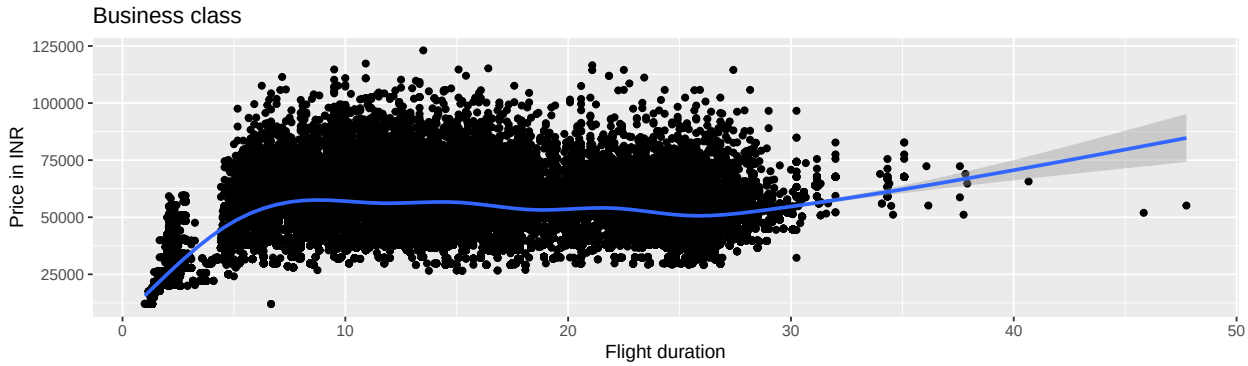


##Price distrbution on the basis of duration of the flight (Economy & Business)

```
ggplot (data = economy_data) +
  geom_point(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price)) +
  labs(y= "Price in INR", x="Flight duration", title = "Economy class") +
  geom_smooth(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price))
```

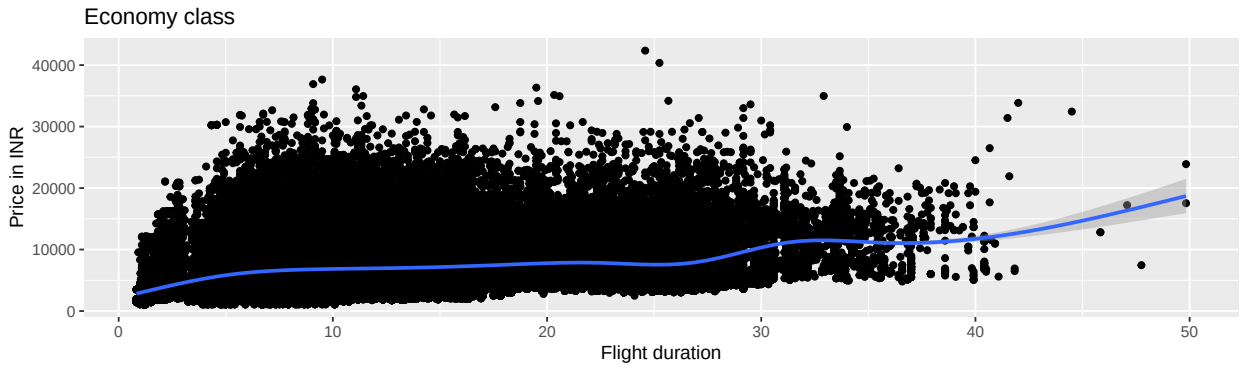



```
ggplot (data = business_data) +
  geom_point(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price)) +
  labs(y= "Price in INR", x="Flight duration", title = "Business class") +
  geom_smooth(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price))
```

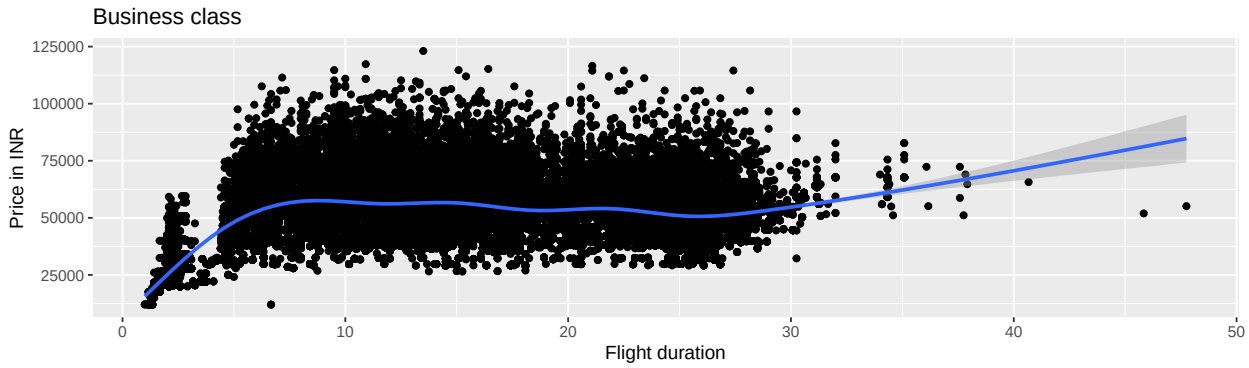


##Price distrbution on the basis of duration of the flight (Economy & Business)

```
ggplot (data = economy_data) +
  geom_point(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price)) +
  labs(y= "Price in INR", x="Flight duration", title = "Economy class") +
  geom_smooth(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price))
```




```
ggplot (data = business_data) +
  geom_point(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price)) +
  labs(y= "Price in INR", x="Flight duration", title = "Business class") +
  geom_smooth(mapping = aes(x=(lubridate::as.duration(time_taken)/3600), y=price))
```



##Model Fitting: Lasso Model

```
#Defining predictor and response variables
y = flight$price
x = model.matrix( ~ ., data =(select(flight,-price, -flight, -source_city, -destination_city, -date)))

#Splitting data
index = sample(nrow(x), floor(0.8 * nrow(x)))
x_train = x[index, ]
y_train = y[index]
x_test = x[-index, ]
y_test = y[-index]

#Fitting the model
model = cv.glmnet(x_train,y_train,alpha =1,folds = 5)
optimal_lambda = model$lambda.min
optimal_model = glmnet(x_train,y_train,alpha =1,lambda = optimal_lambda)
summary(optimal_model)
```

```
##           Length Class     Mode
## a0         1     -none-    numeric
## beta      56     dgCMatrix S4
## df         1     -none-    numeric
## dim        2     -none-    numeric
## lambda     1     -none-    numeric
## dev.ratio  1     -none-    numeric
## nulldev    1     -none-    numeric
## npasses    1     -none-    numeric
## jerr       1     -none-    numeric
## offset     1     -none-    logical
## call       5     -none-    call
## nobs       1     -none-    numeric
```

```
#Extracting model coefficients
coef(optimal_model)
```

```
## 57 x 1 sparse Matrix of class "dgCMatrix"
##                                        s0
## (Intercept)                  5.409335e+04
## (Intercept)                  .
## ...1                        -3.451184e-02
## airlineAirAsia              -3.786460e+02
## airlineGO_FIRST              1.697198e+03
## airlineIndigo                2.065441e+03
## airlineSpiceJet              2.026893e+03
## airlineVistara               4.003859e+03
## departure_timeEarly_Morning  6.323673e+02
## departure_timeEvening        5.402845e+02
## departure_timeLate_Night     1.243225e+03
## departure_timeMorning        7.611241e+02
## departure_timeNight          5.309302e+02
## stops                        5.580514e+03
## arrival_timeEarly_Morning   -7.662068e+02
## arrival_timeEvening          8.420272e+02
## arrival_timeLate_Night       8.466105e+02
## arrival_timeMorning          2.774439e+02
## arrival_timeNight            1.003247e+03
## classEconomy                -5.028543e+04
## duration                     8.089644e+01
## days_left                   -1.244946e+02
## sectorBangalore Delhi       -1.307363e+03
## sectorBangalore Hyderabad   -9.329918e+02
## sectorBangalore Kolkata      2.003224e+03
## sectorBangalore Mumbai       1.785733e+03
## sectorChennai Bangalore      2.070776e+03
## sectorChennai Delhi          2.205982e+03
## sectorChennai Hyderabad      1.291300e+03
## sectorChennai Kolkata        4.411392e+03
## sectorChennai Mumbai         3.819420e+03
## sectorDelhi Bangalore       -3.382507e+03
## sectorDelhi Chennai         -2.456299e+03
## sectorDelhi Hyderabad       -4.007519e+03
## sectorDelhi Kolkata         -7.351220e+02
```

```
## sectorDelhi Mumbai          -4.576695e+03
## sectorHyderabad Bangalore    4.564458e+02
## sectorHyderabad Chennai      8.868337e+02
## sectorHyderabad Delhi       -4.809995e+02
## sectorHyderabad Kolkata      2.196716e+03
## sectorHyderabad Mumbai       1.136048e+03
## sectorKolkata Bangalore      2.905050e+03
## sectorKolkata Chennai        3.061150e+03
## sectorKolkata Delhi          2.373764e+03
## sectorKolkata Hyderabad      1.806567e+03
## sectorKolkata Mumbai         2.568800e+03
## sectorMumbai Bangalore       5.002727e+02
## sectorMumbai Chennai         .
## sectorMumbai Delhi          -3.553126e+03
## sectorMumbai Hyderabad      -1.921756e+03
## sectorMumbai Kolkata         4.526323e+02
## weekdayMonday                8.356079e+02
## weekdaySaturday              5.072201e+02
## weekdaySunday                7.395145e+01
## weekdayThursday              .
## weekdayTuesday               3.030521e+02
## weekdayWednesday             4.146230e+02
```

```
#Counting number of variables removed in the model
(c = coef(optimal_model))
```

```
## 57 x 1 sparse Matrix of class "dgCMatrix"
##                                       s0
## (Intercept)                  5.409335e+04
## (Intercept)                  .
## ...1                        -3.451184e-02
## airlineAirAsia              -3.786460e+02
## airlineGO_FIRST              1.697198e+03
## airlineIndigo                2.065441e+03
## airlineSpiceJet              2.026893e+03
## airlineVistara               4.003859e+03
## departure_timeEarly_Morning  6.323673e+02
## departure_timeEvening        5.402845e+02
## departure_timeLate_Night     1.243225e+03
## departure_timeMorning        7.611241e+02
## departure_timeNight          5.309302e+02
## stops                        5.580514e+03
## arrival_timeEarly_Morning   -7.662068e+02
## arrival_timeEvening          8.420272e+02
## arrival_timeLate_Night       8.466105e+02
## arrival_timeMorning          2.774439e+02
## arrival_timeNight            1.003247e+03
## classEconomy                -5.028543e+04
## duration                     8.089644e+01
## days_left                   -1.244946e+02
## sectorBangalore Delhi       -1.307363e+03
## sectorBangalore Hyderabad   -9.329918e+02
## sectorBangalore Kolkata      2.003224e+03
## sectorBangalore Mumbai       1.785733e+03
```

```
## sectorChennai Bangalore      2.070776e+03
## sectorChennai Delhi          2.205982e+03
## sectorChennai Hyderabad      1.291300e+03
## sectorChennai Kolkata        4.411392e+03
## sectorChennai Mumbai         3.819420e+03
## sectorDelhi Bangalore       -3.382507e+03
## sectorDelhi Chennai         -2.456299e+03
## sectorDelhi Hyderabad       -4.007519e+03
## sectorDelhi Kolkata         -7.351220e+02
## sectorDelhi Mumbai          -4.576695e+03
## sectorHyderabad Bangalore    4.564458e+02
## sectorHyderabad Chennai      8.868337e+02
## sectorHyderabad Delhi       -4.809995e+02
## sectorHyderabad Kolkata      2.196716e+03
## sectorHyderabad Mumbai       1.136048e+03
## sectorKolkata Bangalore      2.905050e+03
## sectorKolkata Chennai        3.061150e+03
## sectorKolkata Delhi          2.373764e+03
## sectorKolkata Hyderabad      1.806567e+03
## sectorKolkata Mumbai         2.568800e+03
## sectorMumbai Bangalore       5.002727e+02
## sectorMumbai Chennai         .
## sectorMumbai Delhi          -3.553126e+03
## sectorMumbai Hyderabad      -1.921756e+03
## sectorMumbai Kolkata         4.526323e+02
## weekdayMonday                8.356079e+02
## weekdaySaturday              5.072201e+02
## weekdaySunday                7.395145e+01
## weekdayThursday              .
## weekdayTuesday               3.030521e+02
## weekdayWednesday             4.146230e+02
```

```
(sum(c==0))
```

```
## [1] 3
```

```
#Extracting pmodel predictions
predicted = predict(optimal_model, x_test)

#Evaluation metrics
auc(y_test, predicted)
```

```
## Area under the curve: 0.8364
```

```
postResample(y_test, predicted)
```

```
##        RMSE    Rsquared         MAE
## 6771.654962    0.911248 4500.919466
```