625-final

Zejia Liu, Jiayuan Xiao

2022-12-17

```
## Rows: 300,153
## Columns: 12
## $ ...1           <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,…
## $ airline        <chr> "SpiceJet", "SpiceJet", "AirAsia", "Vistara", "Vistar…
## $ flight         <chr> "SG-8709", "SG-8157", "I5-764", "UK-995", "UK-963", "…
## $ source_city    <chr> "Delhi", "Delhi", "Delhi", "Delhi", "Delhi", "Delhi",…
## $ departure_time <chr> "Evening", "Early_Morning", "Early_Morning", "Morning…
## $ stops          <chr> "zero", "zero", "zero", "zero", "zero", "zero", "zero…
## $ arrival_time   <chr> "Night", "Morning", "Early_Morning", "Afternoon", "Mo…
## $ destination_city <chr> "Mumbai", "Mumbai", "Mumbai", "Mumbai", "Mumbai", "Mu…
## $ class          <chr> "Economy", "Economy", "Economy", "Economy", "Economy",…
## $ duration       <dbl> 2.17, 2.33, 2.17, 2.25, 2.33, 2.33, 2.08, 2.17, 2.17,…
## $ days_left      <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,…
## $ price          <dbl> 5953, 5953, 5956, 5955, 5955, 5955, 6060, 6060, 5954,…
```

1. Introduction

'Easemytrip' is an internet platform for booking flight tickets, and hence a platform that potential passengers use to buy tickets. Our goal is to analyze/visualize the flight booking dataset obtained from the 'Ease My Trip' website and build a model with higher accuracy to predict flight price. A thorough study of the data will aid in the discovery of valuable insights that will be of enormous value to passengers. A total of 300261 distinct flight booking options was extracted from the site. Data was collected for 50 days, from February 11th to March 31st, 2022. Dataset contains information about flight booking options for flight travel between India's top 6 metro cities.

2. Data processing

1）Transform characters to factors，then convert factors to numbers.

```
##       ...1          airline         flight      source_city departure_time
##  Min.   :     0   1      :  80892   1443   :   3235   :152061   1:47794
##  1st Qu.: 75038   2I     : 16098   1455   :   2741   2:38700   2:46790
##  Median :150076   3I     : 23173   1446   :   2650   3:61343   3:65102
##  Mean   :150076   4I     : 43120   1491   :   2542   4:40806   4: 1306
##  3rd Qu.:225114   5I     :  9011   1478   :   2468   5:46347   5:71146
##  Max.   :300152   6iI27859  1484   :   2440   6:60896   6:48915
##                                    (Other):284037
##          stops        arrival_time   destination_city      class
##  one       :250863   6      :91538   6      :59097   Business: 93487
##  two_or_more: 13286   3      :70323   3      :57360   Economy :206666
##  zero      : 36004   5      :62735   1      :51068
##                      1      :38139   5      :49534
##                      2      :15417   4      :42726
##                      4      :14091   2      :40368
##                      (Other):    0   (Other):    0
##       duration        days_left        price
##  Min.   : 0.83   Min.   :  1   Min.   :  1105
##  1st Qu.: 6.83   1st Qu.:175   1st Qu.:  4783
##  Median :11.25   Median :26    Median :  7425
##  Mean   :12.22   Mean   :26    Mean   : 20890
##  3rd Qu.:16.17   3rd Qu.:38    3rd Qu.: 42521
##  Max.   :49.83   Max.   :49    Max.   :123071
##
```

2)Check up the correlations:

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	duration	days_left	price
airline	1.0000000	0.6459397	-0.0371030	0.0467680	-0.0040438	0.0269987	-0.0386036	-0.0714388	-0.0011578	0.1781643
flight	0.6459397	1.0000000	-0.0094434	0.0715640	-0.1188331	0.0671753	-0.0362484	0.2055017	-0.0004435	0.3058721
source_city	-0.0371030	-0.0094434	1.0000000	-0.0046879	0.0018191	0.0441525	-0.2229348	0.0086194	-0.0035684	0.0045945
departure_time	0.0467680	0.0715640	-0.0046879	1.0000000	-0.0085181	-0.0462824	-0.0017612	0.0843483	-0.0011968	0.0583187
stops	-0.0040438	-0.1188331	0.0018191	-0.0085181	1.0000000	0.0105150	-0.0128462	-0.4738595	-0.0070469	-0.2026202
arrival_time	0.0269987	0.0671753	0.0441525	-0.0462824	0.0105150	1.0000000	-0.0374305	0.0086791	-0.0041914	0.0420427
destination_city	-0.0386036	-0.0362484	-0.2229348	-0.0017612	-0.0128462	-0.0374305	1.0000000	0.0017355	-0.0053228	0.0047950
duration	-0.0714388	0.2055017	0.0086194	0.0843483	-0.4738595	0.0086791	0.0017355	1.0000000	-0.0391569	0.2042224
days_left	-0.0012578	-0.0004435	-0.0035684	-0.0011968	-0.0070469	-0.0041914	-0.0053228	-0.0391569	1.0000000	-0.0919485
price	0.1781643	0.3058721	0.0045945	0.0583187	-0.2026202	0.0420427	0.0047950	0.2042224	-0.0919485	1.0000000

3)Split datasets by training/test 80%/20%:

```
#Split datasets by training/test 80%/20%
df2 <- df1 %>% select(c('airline','flight','source_city','departure_time','stops','arrival_time','destination_city','duration','days_left','price')) %>%
mutate_if(is.factor,as.numeric) %>% drop_na()
set.seed(500)
n <- sample(nrow(df2),nrow(df2)*0.80)
trainset <- df2[n,]
testset <- df2[-n,]
dim(trainset)
```

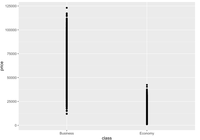
```
## [1] 240122     10
```

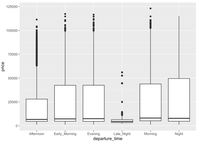
```
dim(testset)
```

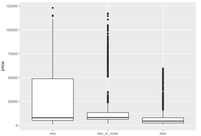
```
## [1] 60031    10
```

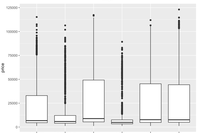
2. Data Visualization

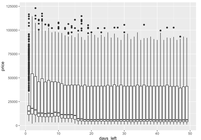
We have already analyzed the correlation earlier.By ggplot, we can visually observe the relationship between each variable in the data we collected and the price of airline tickets. From there, it can help consumers to simply determine and predict the price of airline tickets according to airline, cities, and departure time etc.

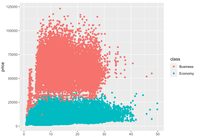


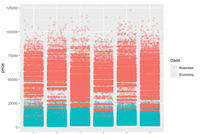




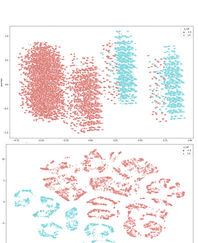








Dimension Reduction



As we can see from the result of PCA and tSNE, the data can be divide into two group, which indicates that our machine learning algorithm might have the potential to achieve high accuracy.

3.Model and Result

We applied three types of methods to predict the flight ticket price. Linear regression and its extensions are classic approaches in mechine learning algorithm. Support vector regression uses the same idea as SVM. XGBoost is a efficient algorithm based on the gradient boosting framework. Below is the model comparison result of these models.

Model	RMSE	R2_score
Linear regression	7005.015	0.905
Lasso regression	7005.006	0.905
Ridge regression	7005.012	0.905
Elastic net regression	7005.008	0.905
Support vector regression	14874.945	0.57
XGBoost	3368.904	0.978

As shown in the chart, the results of linear regression, Lasso regression, Ridge regression and Elastic net regression have no significant differences. This might suggest that the dimension of variables in this dataset is not too high. The R2 score of the support vector regression is low, which shows that SVR might not suitable for this dataset. As one of the most popular prediction algorithms in kaggle, we can see from the table that XGBoost outperforms other algorithms.

4.Challenge and future work

Since the dataset has about 300,000 observation, the main challenge of this project is the computational challenge. Algorithms like t-SNE and SVR can be very time-consuming in this case. Next step, we will try to run the models on gpu to improve the efficiency. Besides, we will also try other ML algorithms to see if they can achieve higher accuracy.