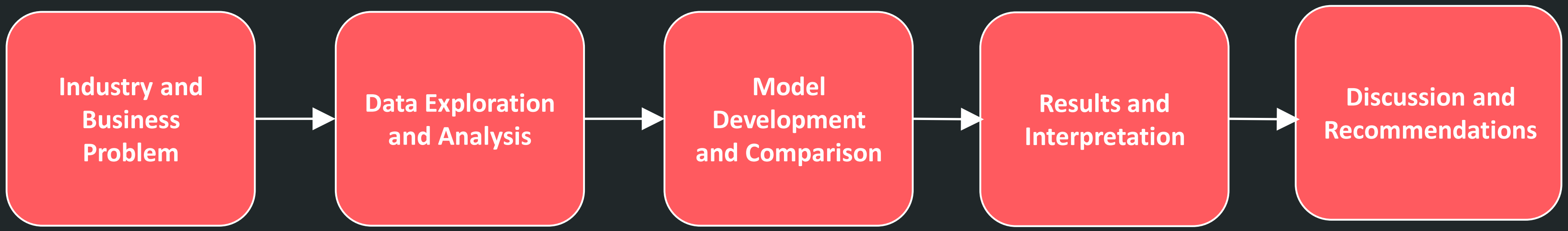
airbnb

Predicting the next Holiday destination

Introduction



Airbnb is an online platform that allows their users to browse through a collection of residences spread over 65,000 cities in more than 190 countries across the world.

Our dataset was acquired from Kaggle and consisted of variables such as: Date of Account Creation, Date of First Booking, Gender, Age, Signup Method, Language, Country Destination and many others.

The Challenge at Hand

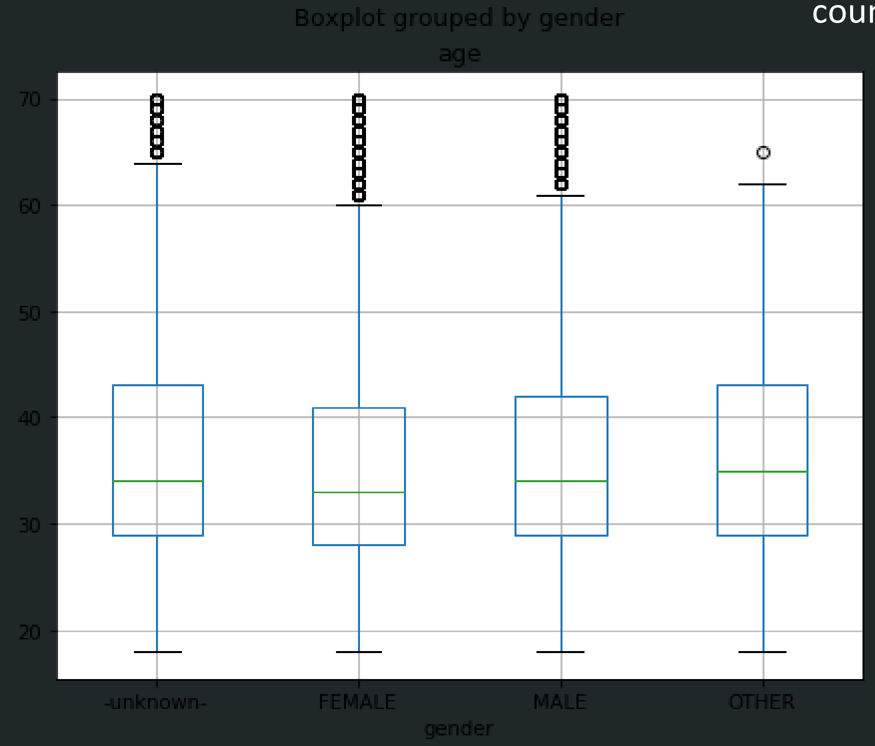
Is there a method we can use to predict where the user will go given the variables in our dataset? And if so, how can we leverage that information to help the business grow and create actionable insights?

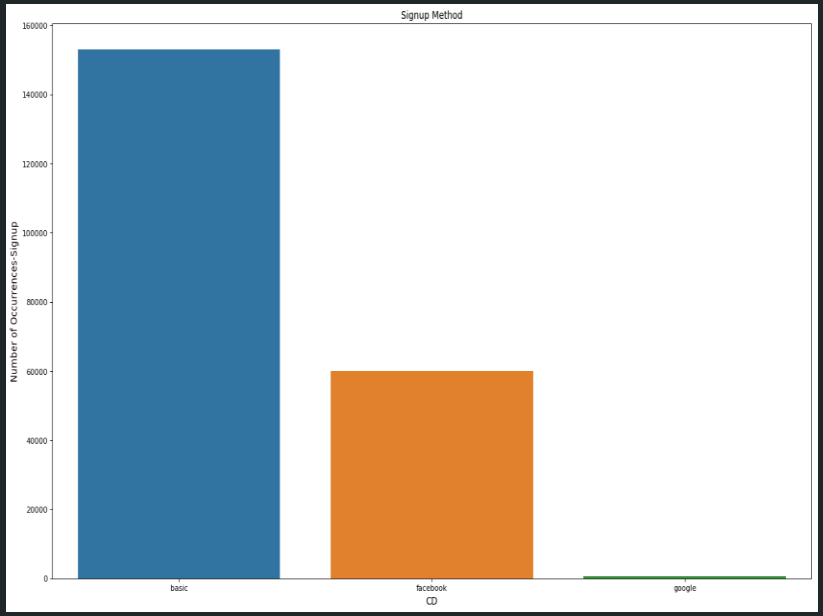
Exploratory Analysis

Taking a deeper look and Understanding our Data

Data provided by Airbnb on Kaggle → **Out of 213k data points with 16 features we found a lot of missing values for 4 features** → **Interesting Data Points were for age where lot of values existed for people below 18** → **Trying to find some predictors via initial analysis**

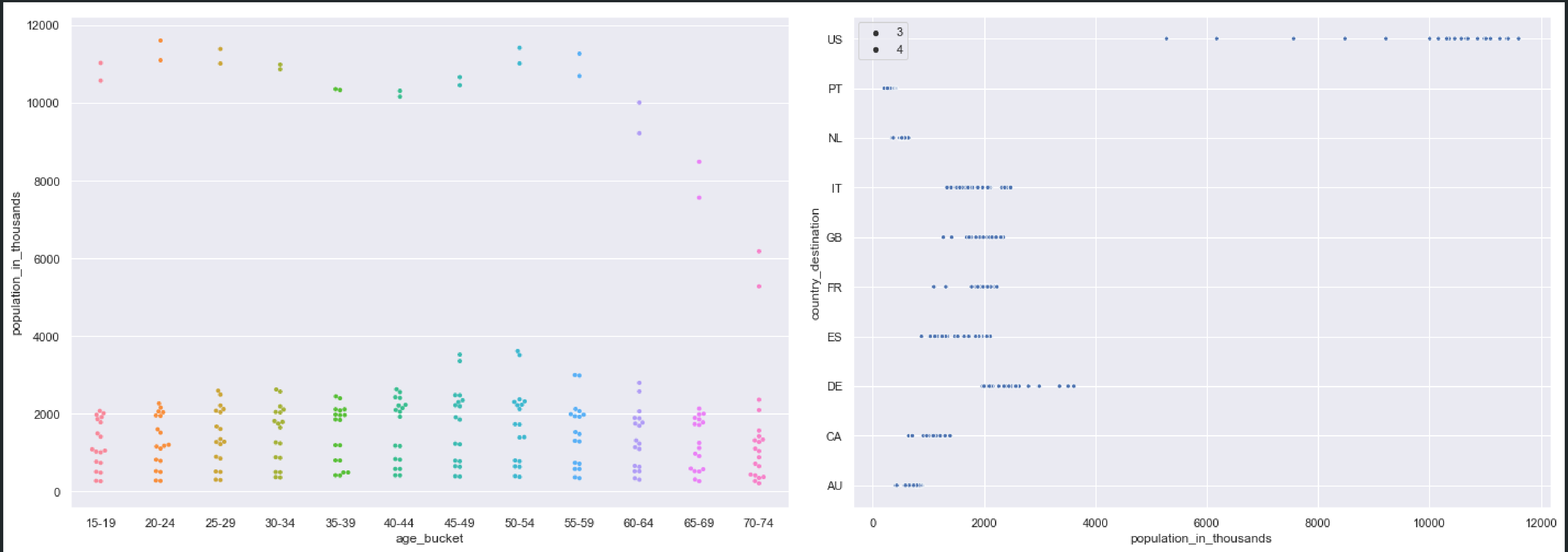
Comparing Gender versus Age group, we observed that the median age group is around 32 to 35 but are different distributions & We observed that basic signup method was used for sign up for most of the countries





Exploratory Analysis

Plotting the no. of people based on age range, we can see that the age group of 20-24,25-29 & 50-54 have the highest no. of people interested in going for a vacation; Plotting the countries most people are interested in going to for a vacation; US clearly is the winner here followed by Denmark & Italy



Exploratory Analysis

Conducting a t-test with the below Hypothesis:
H0 = The avg age of people wanting to go to the US is approximately equal to the ones wanting to go to Denmark
H1= The avg age of people wanting to go to the US is not at all equal to the ones wanting to go to Denmark

```
In [19]: # Conducting a t test on two independent samples
         from scipy.stats import ttest_ind_from_stats
         ttest_ind_from_stats(mean1=dfus_age_m, std1=dfus_age_sd, nobs1=62376,
                              mean2=dfne_age_m, std2=dfne_age_sd, nobs2=1061, equal_var = False)

Out[19]: Ttest_indResult(statistic=-1.6026026763030288, pvalue=0.10931213166343735)

In [ ]:  #Based on the above p value of 10% we can clearly reject our null hypothesis which states that the avg age of people wanting
         #to go to the US is equal to the ones wanting to go to Denmark
```

Conducting a ANOVA test with the below Hypothesis:
#H0= The signup method does not relate anyways to the age of the person

```
#Performing a one way ANOVA
F, p = stats.f_oneway(df1, df2, df3)
print('F statistic = {:5.3f} and probability p = {:5.3f}'.format(F, p))

F statistic = 207.185 and probability p = 0.000

#Interpretation of results:
#As p < a (0.05) we state that we have a main interaction effect. This simply means that amongst group comparison
#identifies statistically significant differences. However, this result does not identify the sample pair (or pairs)
#which cause this significance.
```

Feature Selection & Engineering

Multiple features were tested with Chi^2 to see their impact on the target variable → Dropped the features that were irrelevant & with lots of null values → Collapsed multiple categories in features for better modelling → Converting each categorical variable to numerical for modelling

```
#Creating a contigency table for Chi Square test:
#H0= We believe there is no relationship between signup method and country of destination (are independent of each other)
#H1 = These two variable are not independent of each other
contingency_table = pd.crosstab(
    df['signup_method'], df['country_destination'])
contingency_table
#Each cell in this table represents a frequency count.
```

country_destination	AU	CA	DE	ES	FR	GB	IT	NDF	NL	PT	US	other
signup_method												
basic	282	738	507	1038	2387	1127	1300	21293	386	91	30899	4748
facebook	141	309	317	623	1214	578	652	33598	194	62	16315	2572
google	0	1	1	1	3	0	2	57	0	1	65	9

```
from scipy import stats
stats.chi2_contingency(contingency_table)[:3]
#With a p-value < 0.05 , we reject the null hypothesis. There is a strong relationship between
#'signup method' and the 'country of destination' column, we can see that these two variables are not
#independent of each other.
```

```
(8616.825481629532, 0.0, 22)
```

```
#Creating a contigency table for Chi Square test:
#H0= We believe there is no relationship between the device used to browse/signup and country of destination
#(are independent of each other)
#H1 = These two variable are not independent of each other
contingency_table = pd.crosstab(
    df['first_device_type'], df['country_destination'])
contingency_table
#Each cell in this table represents a frequency count.
```

9]:

country_destination	AU	CA	DE	ES	FR	GB	IT	NDF	NL	PT	US	other
first_device_type												
Android Phone	0	11	4	9	14	2	11	690	2	0	432	65
Android Tablet	0	6	3	8	18	1	11	305	5	2	223	41
Desktop (Other)	3	17	12	7	17	5	6	317	4	0	304	43
Mac Desktop	224	525	452	853	1966	932	999	23560	305	84	23323	3363
Other/Unknown	8	16	7	28	55	24	17	2667	9	2	930	138
SmartPhone (Other)	0	0	0	0	1	0	1	21	0	0	15	1
Windows Desktop	128	379	253	551	1081	533	631	18638	166	51	15511	2662
iPad	32	48	45	102	244	109	161	3478	56	7	2666	472
iPhone	28	46	49	104	208	99	117	5272	33	8	3875	544

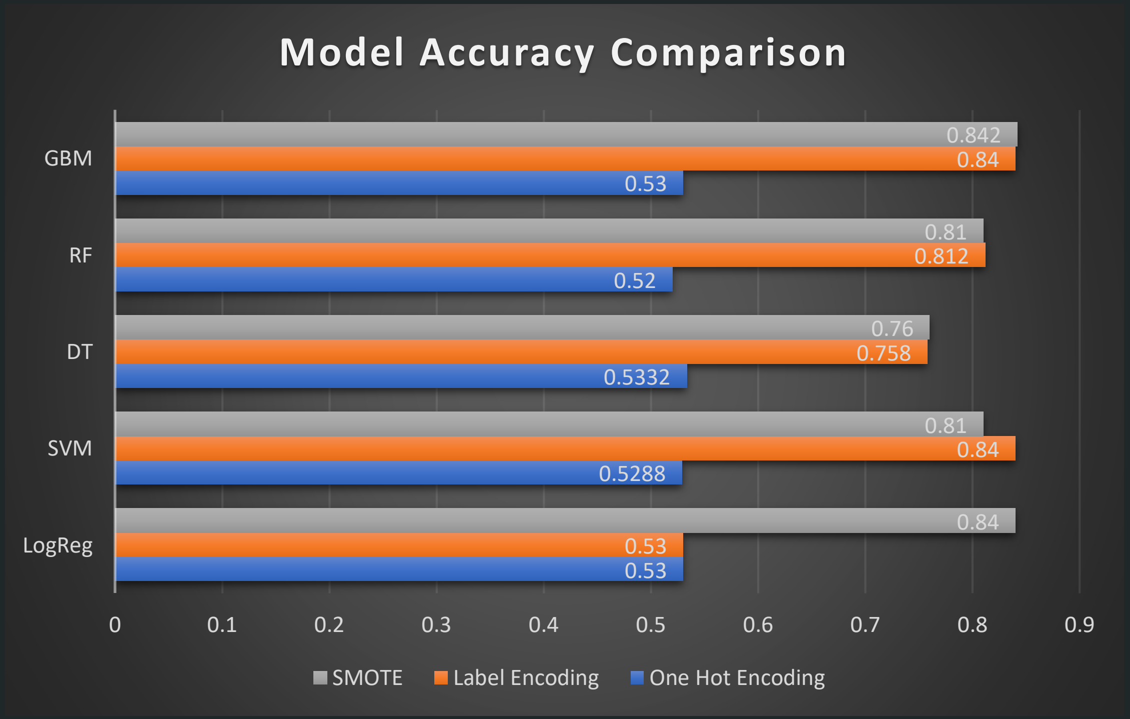
```
from scipy import stats
stats.chi2_contingency(contingency_table)[:3]
#With a p-value < 0.05 , we  reject the null hypothesis. There is visible relationship between
#'first device type' and the 'country of destination' column, we can see that these two variables are not
#independent of each other.
```

0]:

```
(1662.6538804271956, 5.641255919580156e-289, 88)
```

Modelling

We had to repeat the modelling exercise three times based on multiple encoding → Please note hyperparameter tuning was also tried but did not impact the accuracy → Had to apply SMOTE to balance the data post both the encoding → Although post SMOTE modeling did give better scores the data was still imbalanced




```
118]:  xgb.fit(Xn_train, yn_train)
       y_pred=xgb.predict(X_test)
       # Model Accuracy, how often is the classifier correct?
       print("Accuracy:",metrics.accuracy_score(y_test, y_pred))

       Accuracy: 0.8426492856672592

119]:  print(classification_report(y_test, y_pred))

                     precision    recall  f1-score   support

                AU       0.00      0.00      0.00        99
                CA       0.00      0.00      0.00       256
                DE       0.00      0.00      0.00       202
                ES       0.00      0.00      0.00       423
                FR       0.00      0.00      0.00       865
                GB       0.00      0.00      0.00       440
                IT       0.00      0.00      0.00       468
               NDF       1.00      1.00      1.00     13780
                NL       0.00      0.00      0.00       124
                PT       0.00      0.00      0.00        38
                US       0.71      1.00      0.83     11831
             other       0.50      0.00      0.00      1852

          accuracy                           0.84     30378
         macro avg       0.18      0.17      0.15     30378
      weighted avg       0.76      0.84      0.78     30378
```

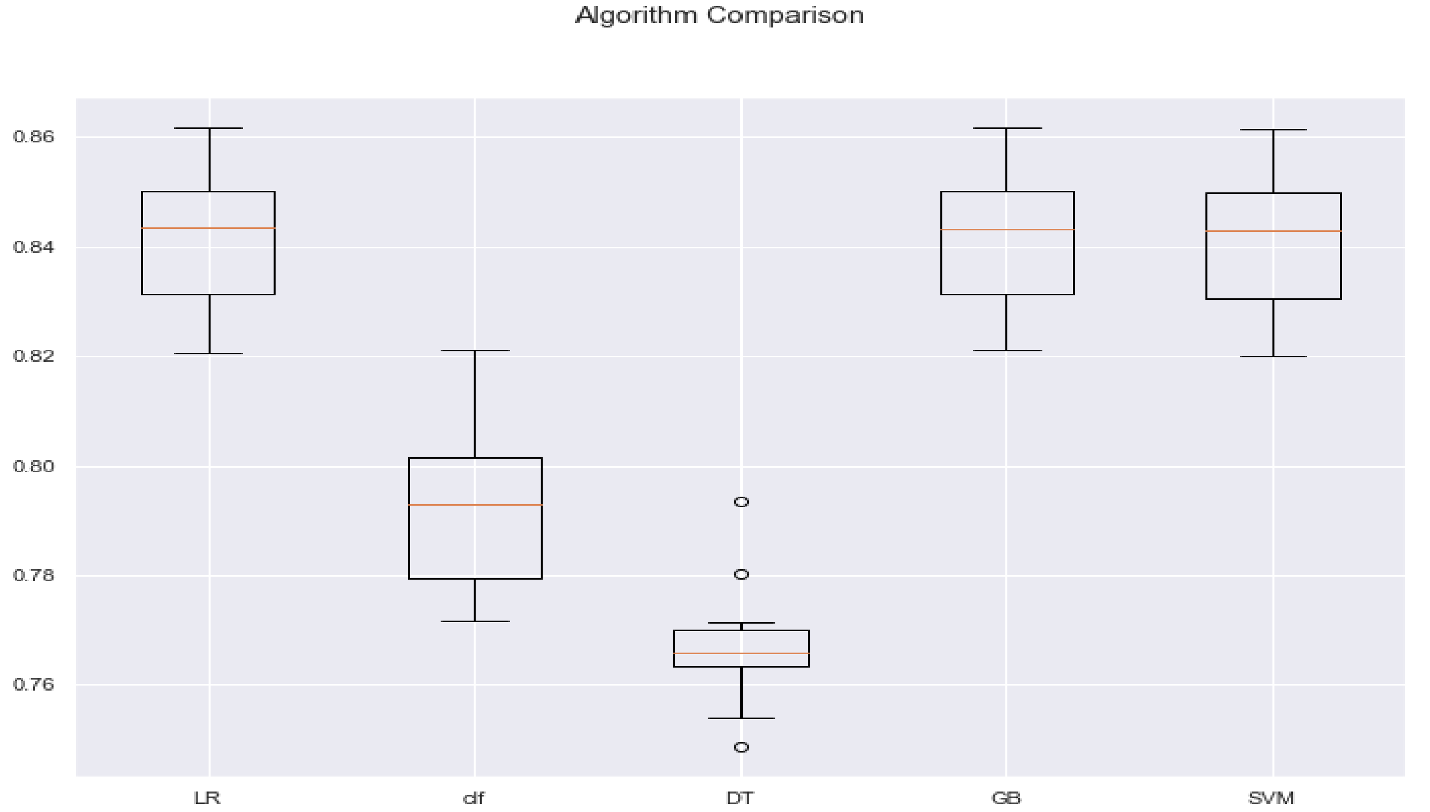
Re-Modelling

Since the data was imbalanced, we combined countries with lesser data to the others category & retry the model with 3 classes- US, NDF & Others

The models post collapsing the target variable showed good and trustworthy results

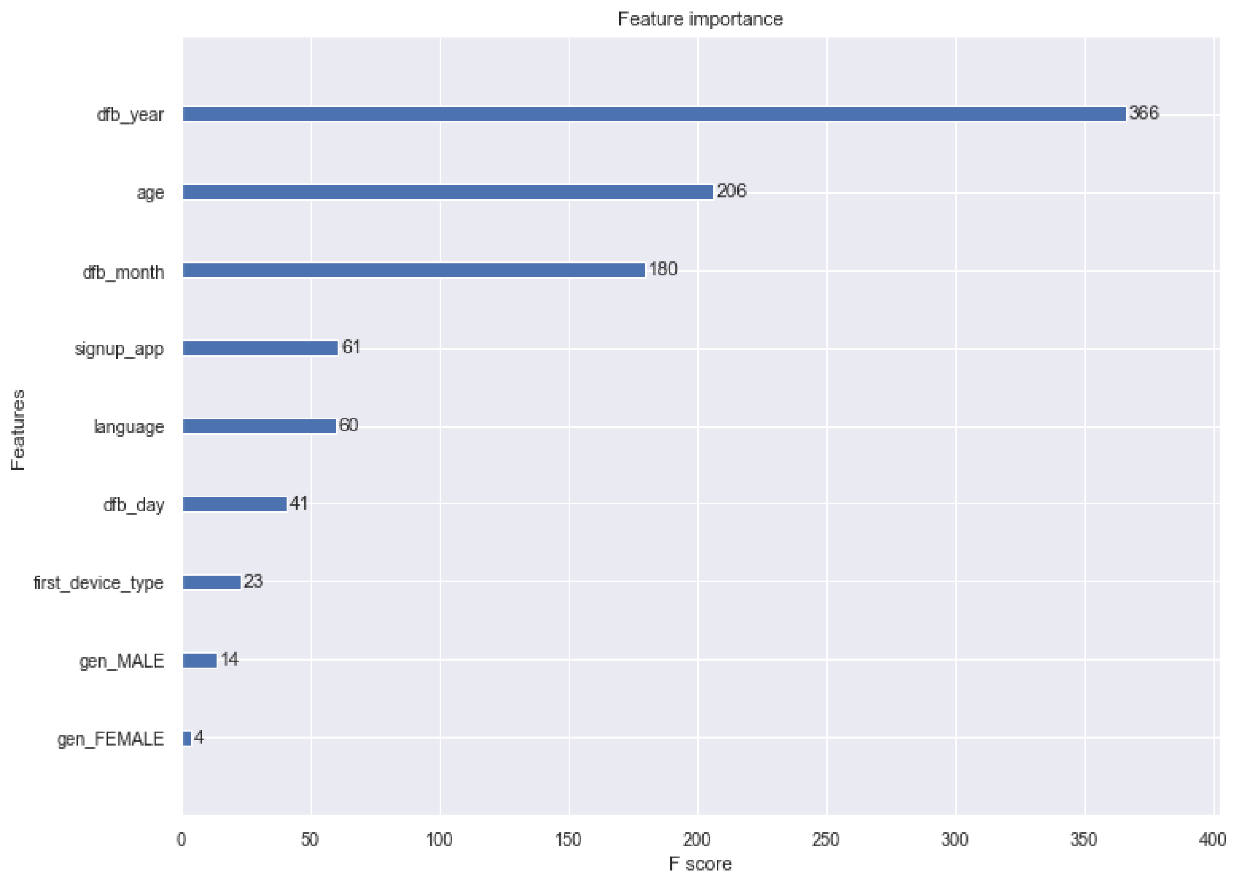
Our best models are GB & LR, but GB has a better *SD.*

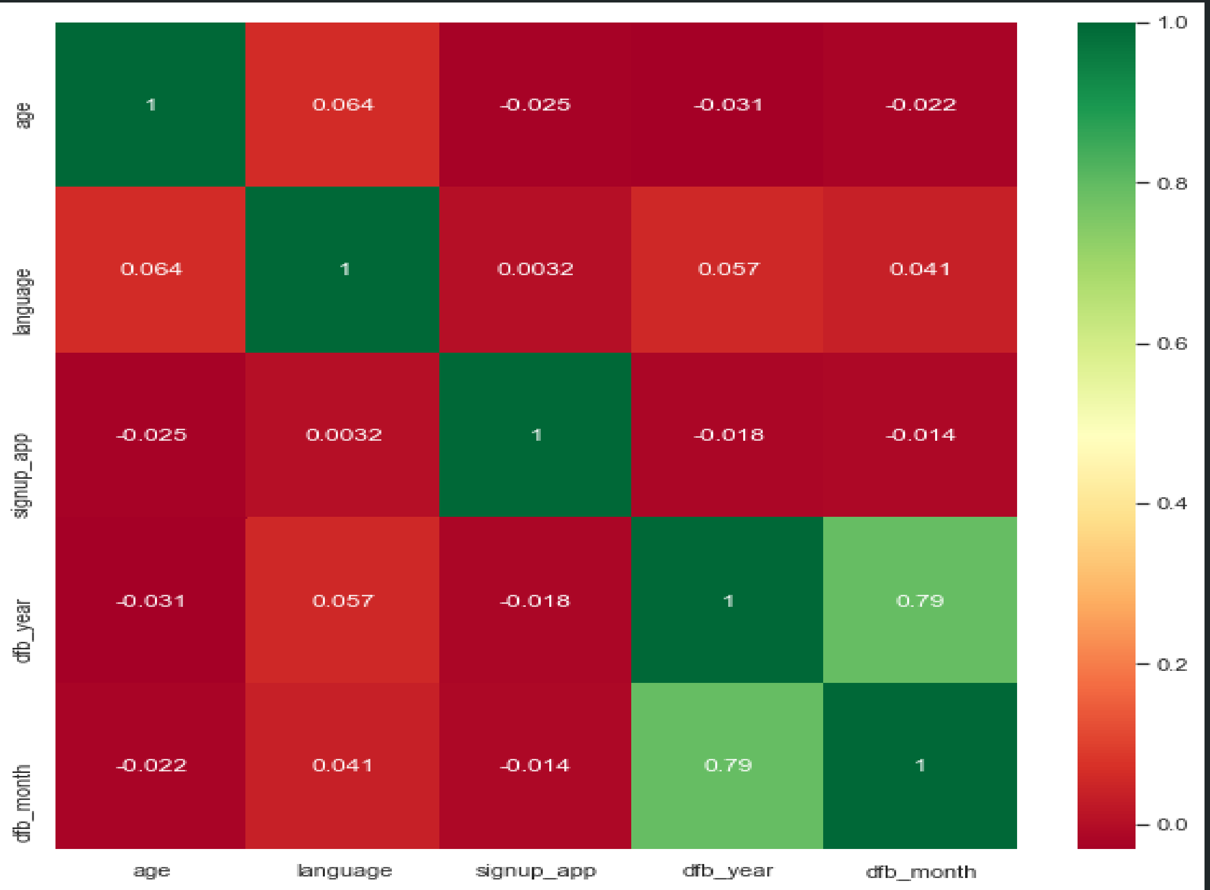
```
In [305]: ▶ updated_countries.value_counts()

Out[305]: NDF       54948
          US        47279
          Others    19284
          Name: country_destination, dtype: int64
```



Identifying best predictors

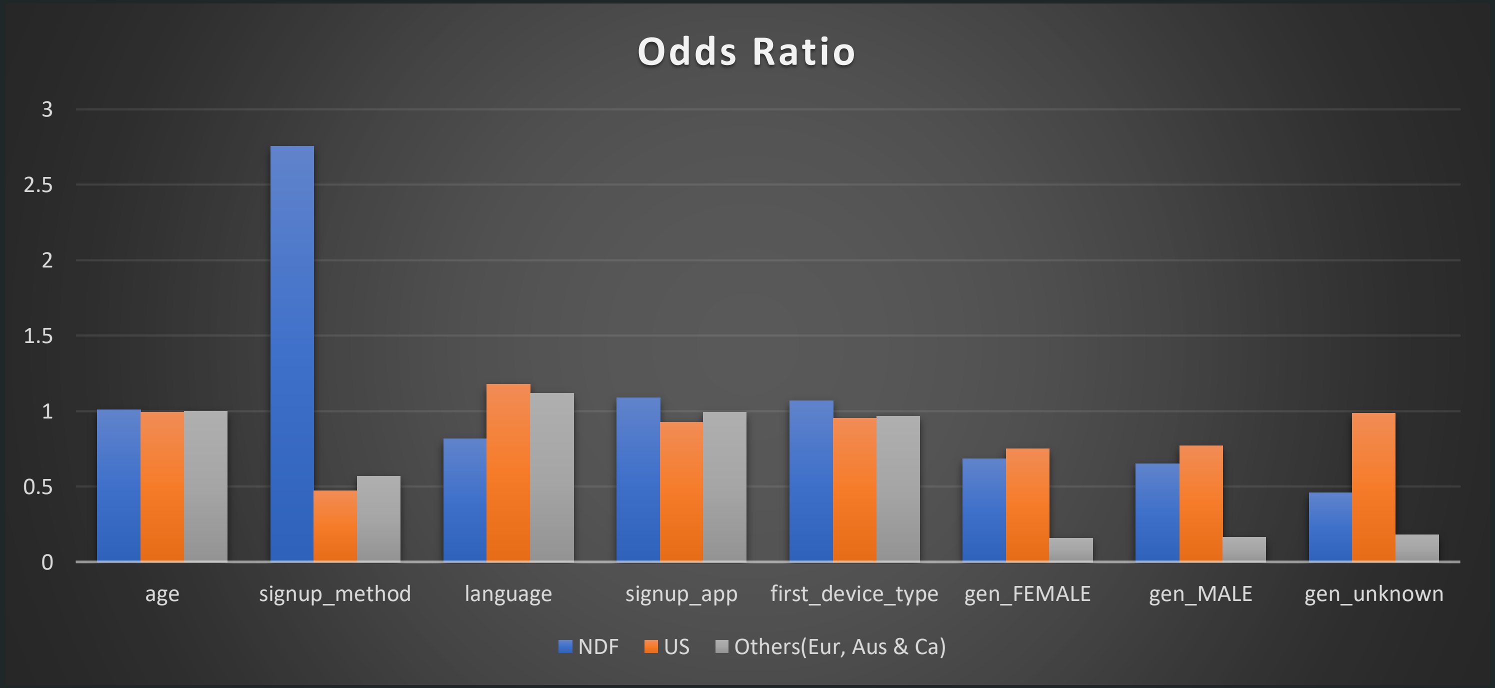
We checked for the best predictor variable in our GB model; post which we also wanted to check if there is any correlation among the top five predictors. The only high correlation is between date of first booking year & month, apart from that there are no high correlations to notice





Country Specific Predictors

Based on the odds ratio we can predict the probabilities of each feature contributing to choosing a destination. For example signup method is the biggest impactor for NDF whereas language is for the US & Other countries. We can always look for the top three predictors for each country and align our marketing proposition accordingly based on the category inside each feature



Features	NDF	US	Others(Eur, Aus & Ca)
age	1.00978	0.990863	0.999007
signup_method	2.756993	0.472126	0.569062
language	0.814498	1.178149	1.120363
signup_app	1.088207	0.924717	0.99234
first_device_type	1.070269	0.952315	0.966298
gen_FEMALE	0.685479	0.753881	0.158632
gen_MALE	0.653397	0.770859	0.166571
gen_unknown	0.460004	0.986122	0.180535

Recommendations

Personalized promotions based on highest probability of destination country booking

Use promotions based on probabilities to move customers from non-booking to booking

In case a user has closer probability values of going to all countries then we recommend offering generic promotion

THANK YOU