Airbnb New User Bookings Project - Report

Objective:

Airbnb's business covers 34,000+ cities across 190+ countries. Thus, being able to accurately identify where new users are heading to is important as it allows Airbnb to recommend rentals that suit customers' needs and as a result, decrease the average time to first booking and improve the site's booking rate overall.

Research Question: Predict which country new users will book their first trip based on users' demographic data, web session records, and some summary statistics of different countries.

Data:

The datasets for this project are available on Kaggle (https://www.kaggle.com/c/airbnb-recruiting-new-user-bookings)
These datasets are:

- train_users.csv: training data. The file includes information related to Airbnb accounts such as when the user signed up/ made his/her first booking, sign-up flow, language preference, etc.
- test_users.csv: testing data. It contains the same fields as train_users.csv's except that test_users.csv does not include the outcome variable which is the destination country for the first booking.
- age_gender_bkts.csv: It includes different countries' age/ gender splits.
- countries.csv: It includes geographic information of different countries.
- sessions.csv: It contains users' web session log.

Data Wrangling:

For train_users_2 & test_users datasets:

- Parse datetime variables 'date_account_created', 'date_first_booking' and 'timestamp_first_active' to pandas datetime objects.
- Replace '-unknown-' with numpy NaN.
- 0.01% accounts were created after the first trip was booked ('date_account_created' > 'date_first_booking'). According to Airbnb's terms & conditions, users need to register for an Airbnb account first before booking and so I replace the 'date_account_created' with 'date_first_booking' for these observations.
- 1% account holders are > 100 years old while 0.068% are < 18 years old. Since Airbnb requires users to be at least 18 years old in order to use the site (https://www.airbnb.com/terms), I replace all these < 18 years old with 18. I also replace those > 100 years old with 100.
- Create variables that capture the year, month, day, week and day of week of 'date_account_created', 'date_first_active'.
- We have more than 40% NaNs for age & gender. Use machine learning to predict these missing values from other variables because these two variables are need to join the age_gender_bkts dataframe. Create a binary indicate to record if their original values

are NaNs since the fact that these values are missing might tell us something about the data.
- Also use machine learning to fill out missing values for language.
- Drop 'date_first_booking' since this feature is not available in the test set (all NaN).
- Do one-hot encoding for categorical variables. For each categorical variable, include an indicator that records if the value of the observation is missing.

For age_gender_bkts:
- Transform the dataframe from long to wide format for merging with train & test data.

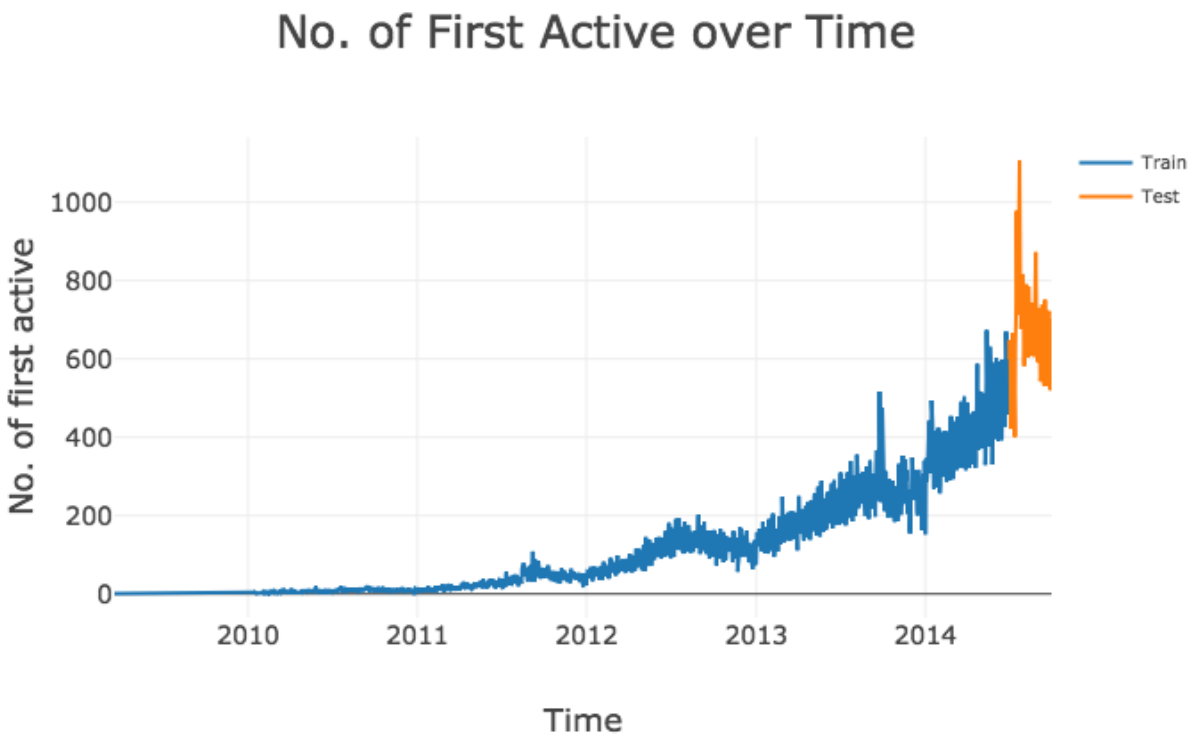
For countries datasets:
- Variable 'destination_language ' is renamed 'destination_language'.
- Values of 'destination_language' is mapped to the 'language' variables in train & test sets.

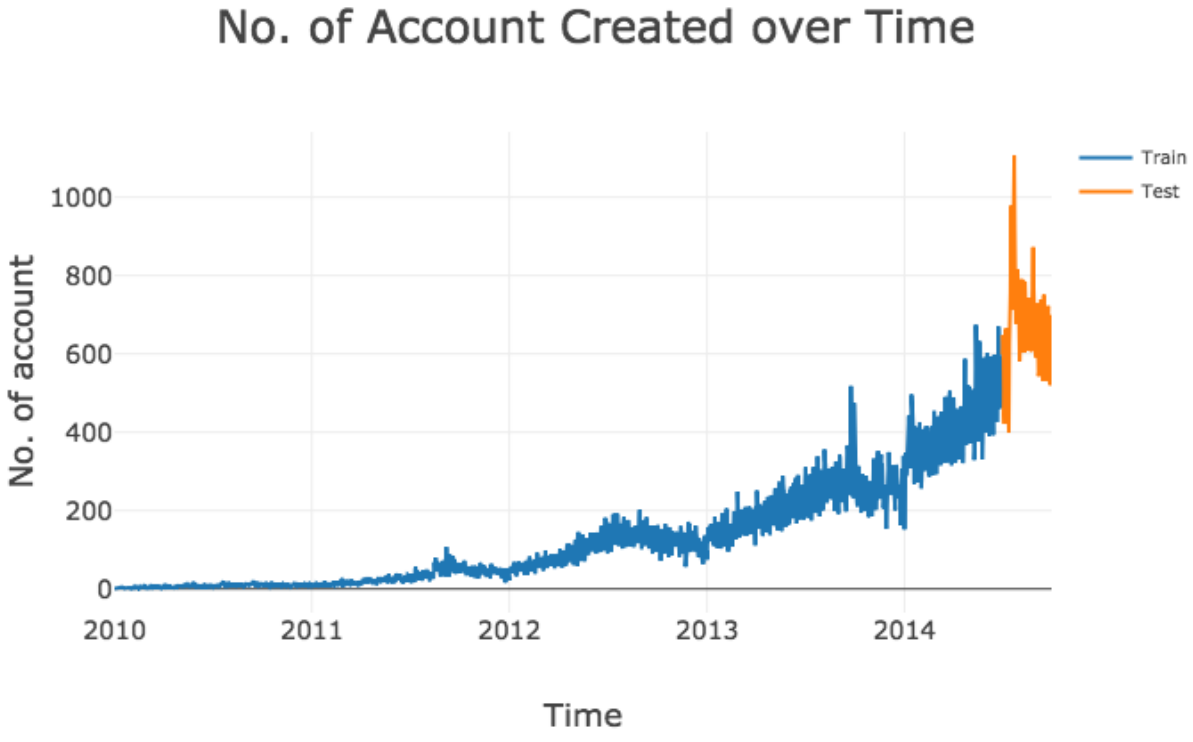
For session dataset:
- Replace '-unknown-' with numpy NaN.
- Observations with missing user_ids are dropped since it is the key variables for merging with the train/ test data.
- 1.7% sessions with session time > 200,000 seconds (55.56 hours). Leave them as is and will probably try log transformation during modeling.
- Drop observations with missing session time as they probably are not legit sessions
- Create a dataframe (df_session2) that records for each user:
    - Total number of actions taken
    - Total number of unique Actions, ActionTypes, Action_Details, Devices
    - Sum, mean, min, max, median, s.d., skewness, kurtosis of seconds_elapsed
    - Number and proportion (sum & mean) of each type of action taken, e.g. 'lookup', 'search_results'

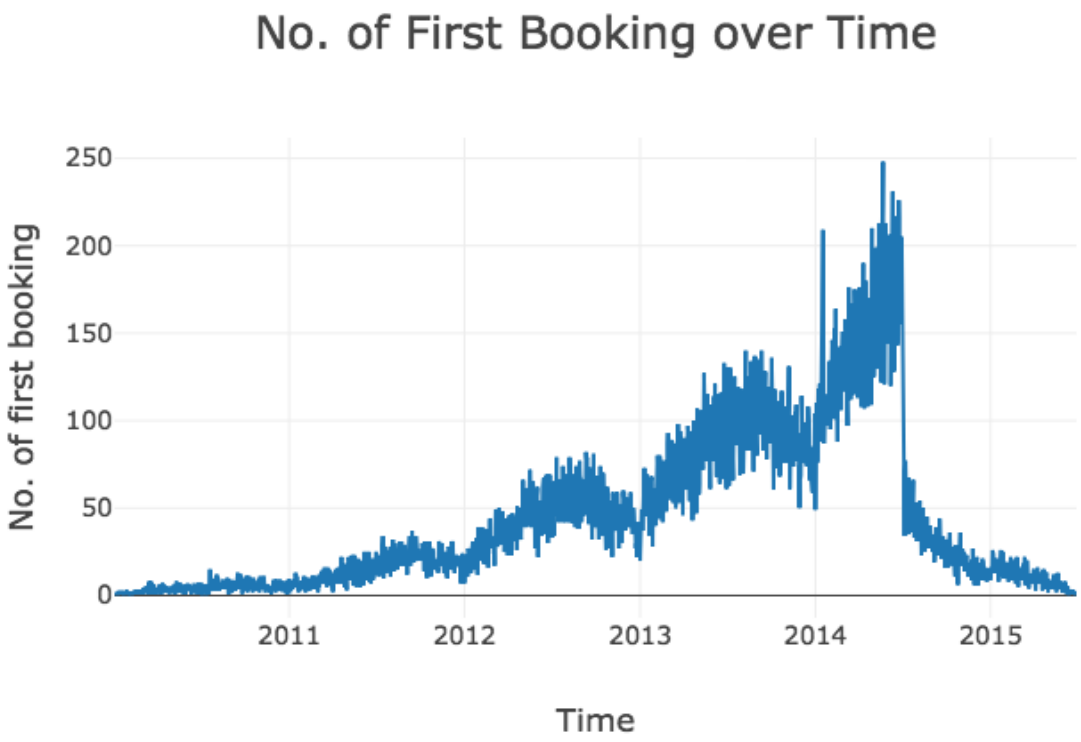
Merging datasets:
- Merge train_users_2 & test_users datasets with age_gender_bkts using age and gender.
- For the merged datasets, create a variable that records if the user's preferred language matches the language of each country using the information from the countries dataframe
- Join the merged datasets with the session data (df_session2) using user_id

After merging the datasets:
- Some user_id appear in training & test datasets but not in session data and so we have lots of NaNs for variables related to sessions after merging the dataset. Replace these NaNs with 0.

Exploratory Analysis:

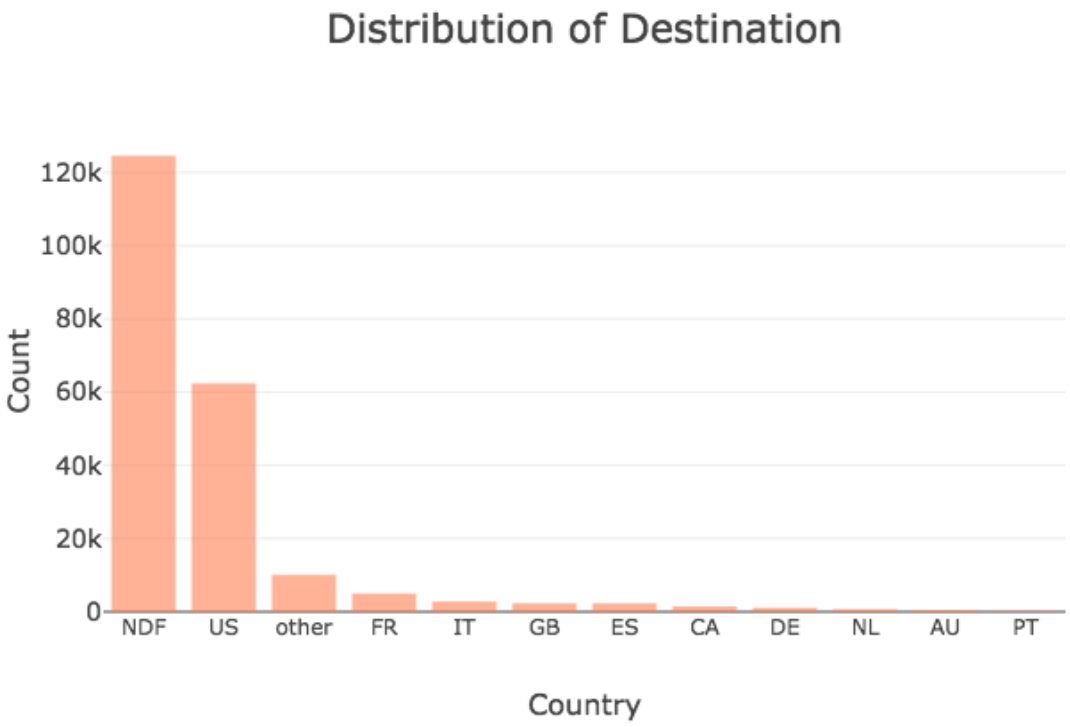




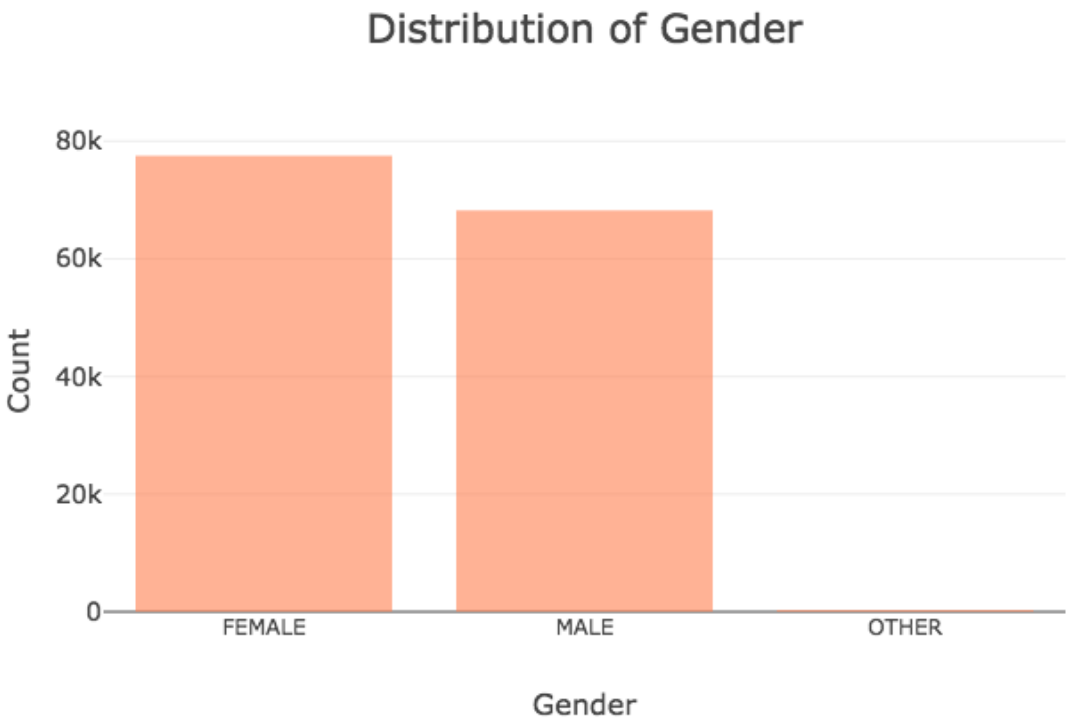


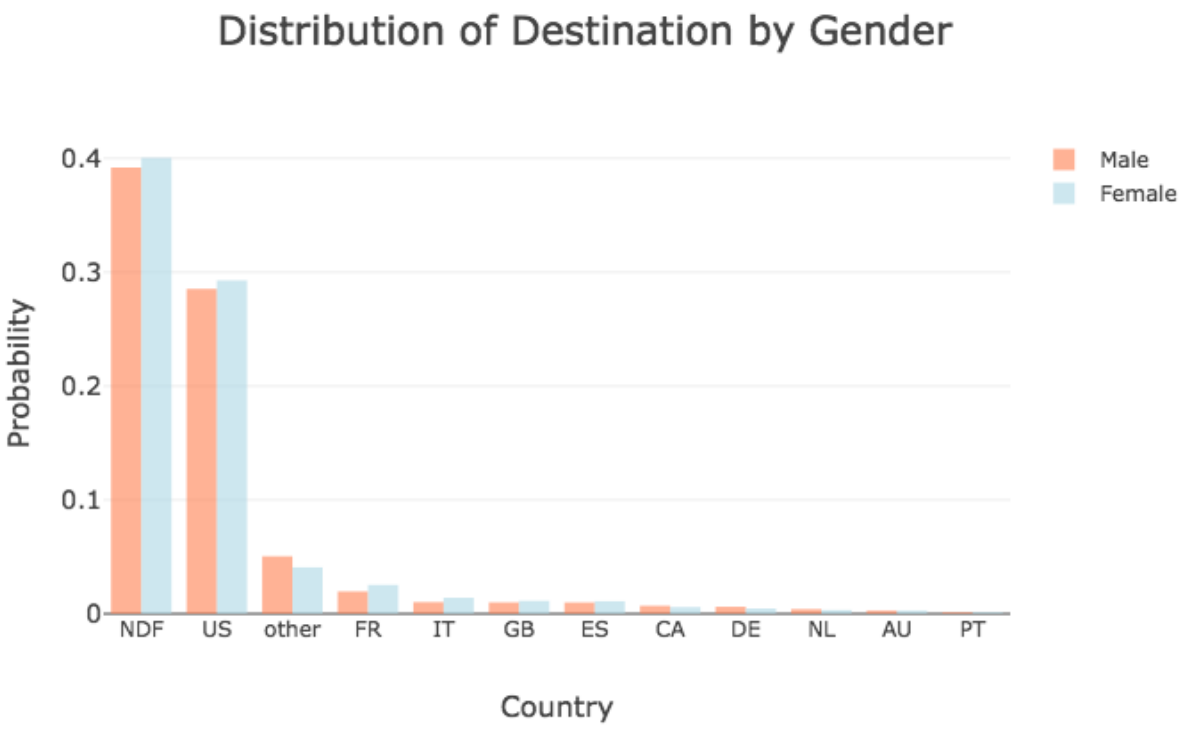
- Training set contains users with first active date and account creation date before 2014-07-01 while test set contains those with first active date and account creation date on or after 2014-07-01.
- Seems like in mid July 2014, there were an influx of new users trying and signing up for Airbnb without actually booking their first trip.

Let's look into these Airbnb users in terms of who they are and where they booked their first trip.

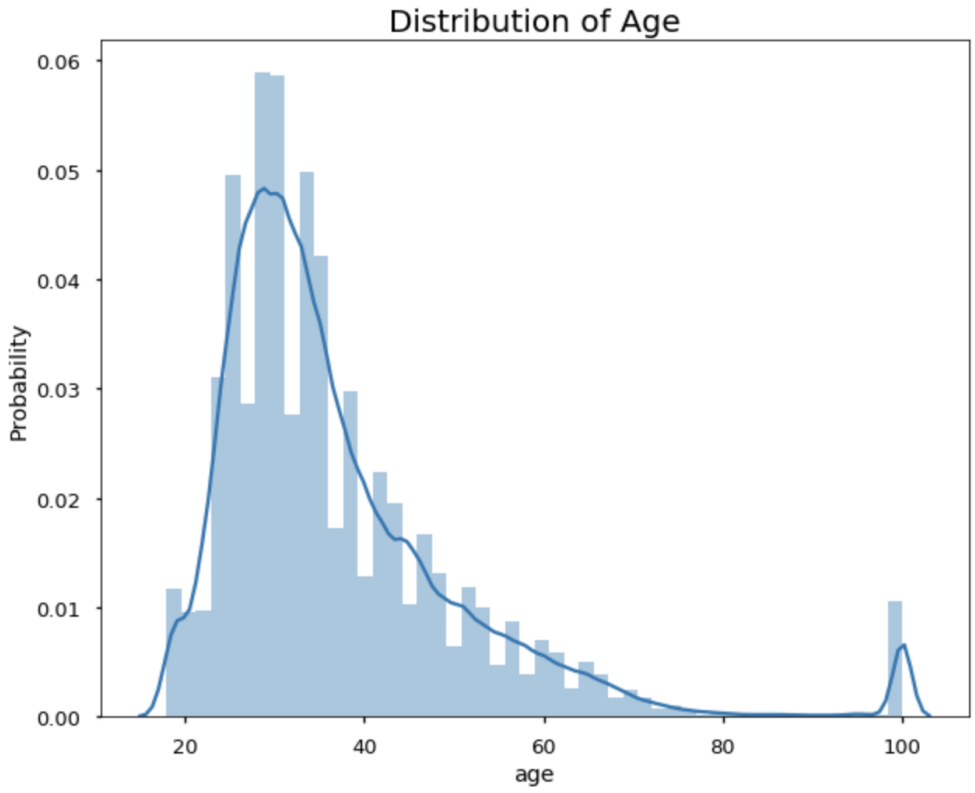


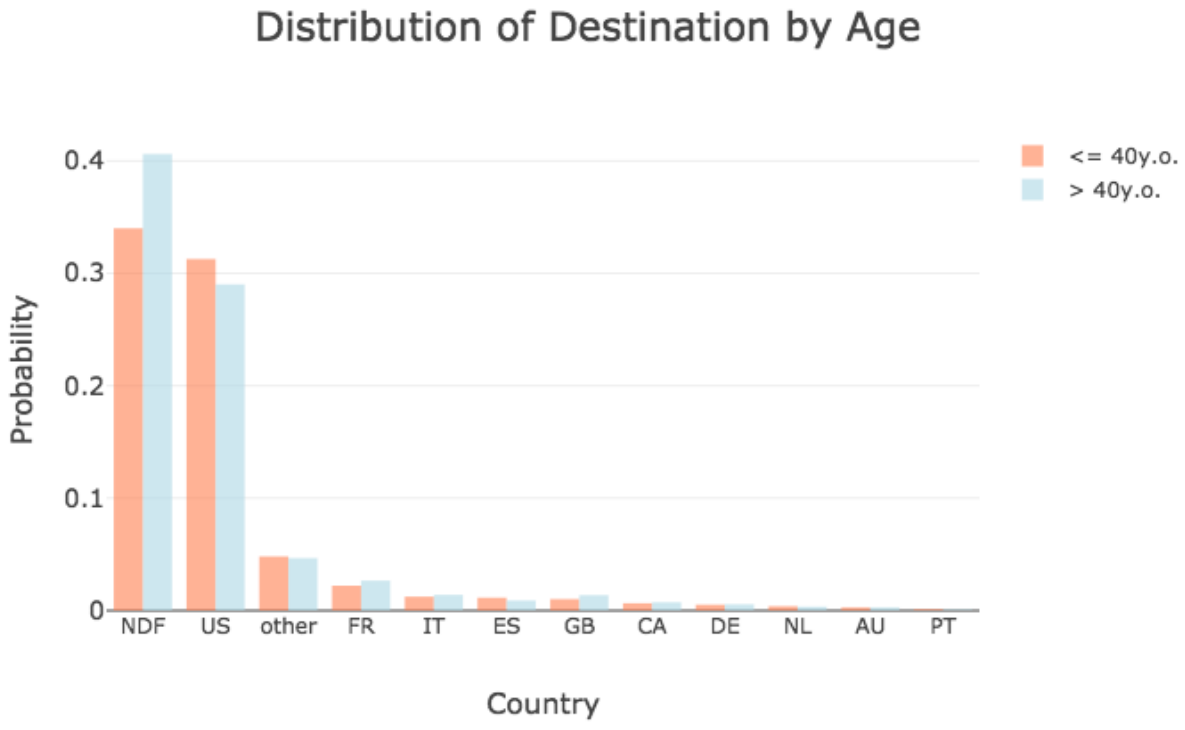
- Most users did not end up booking a trip; for those who ended up booking, most of them booked a trip to the U.S.



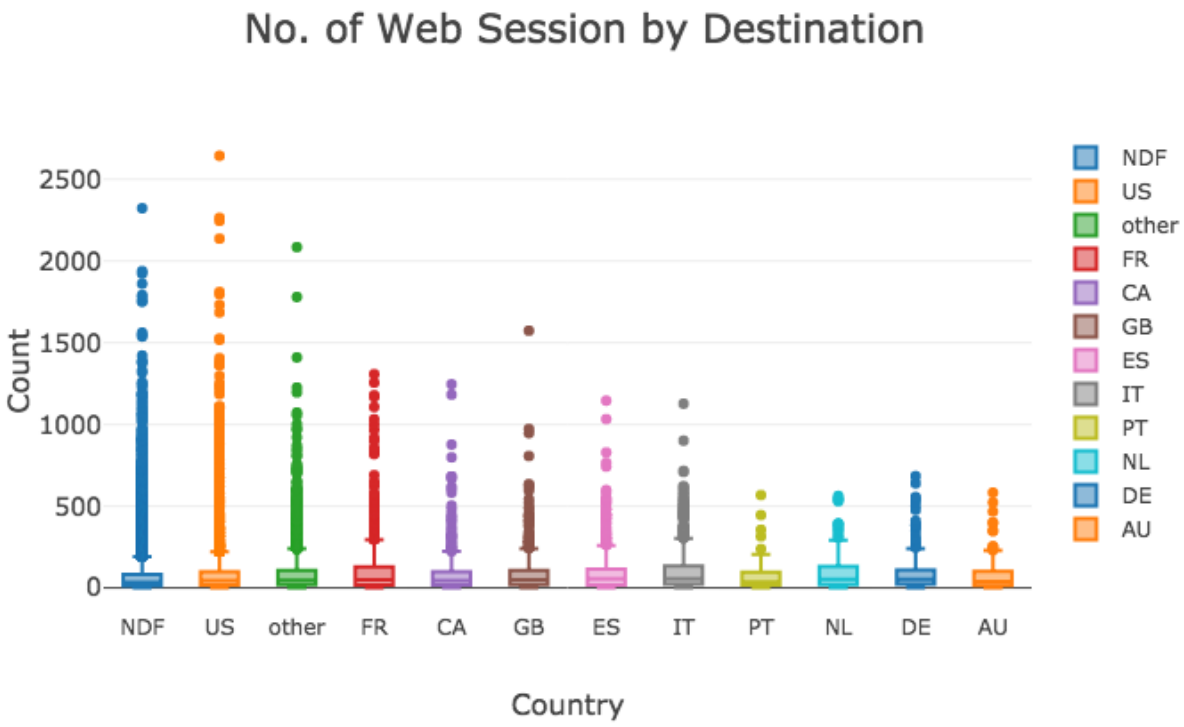


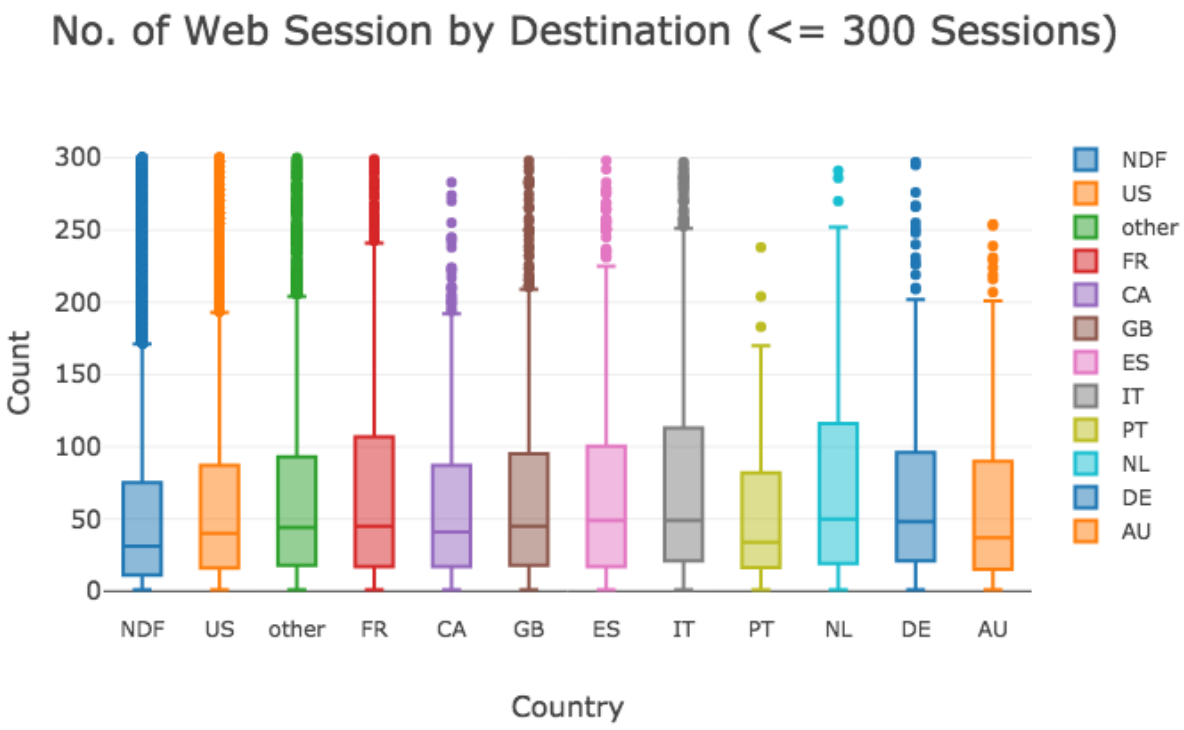
- We have slightly more female users than male. However, male and female users did not seem to show different preference when picking their first destination.

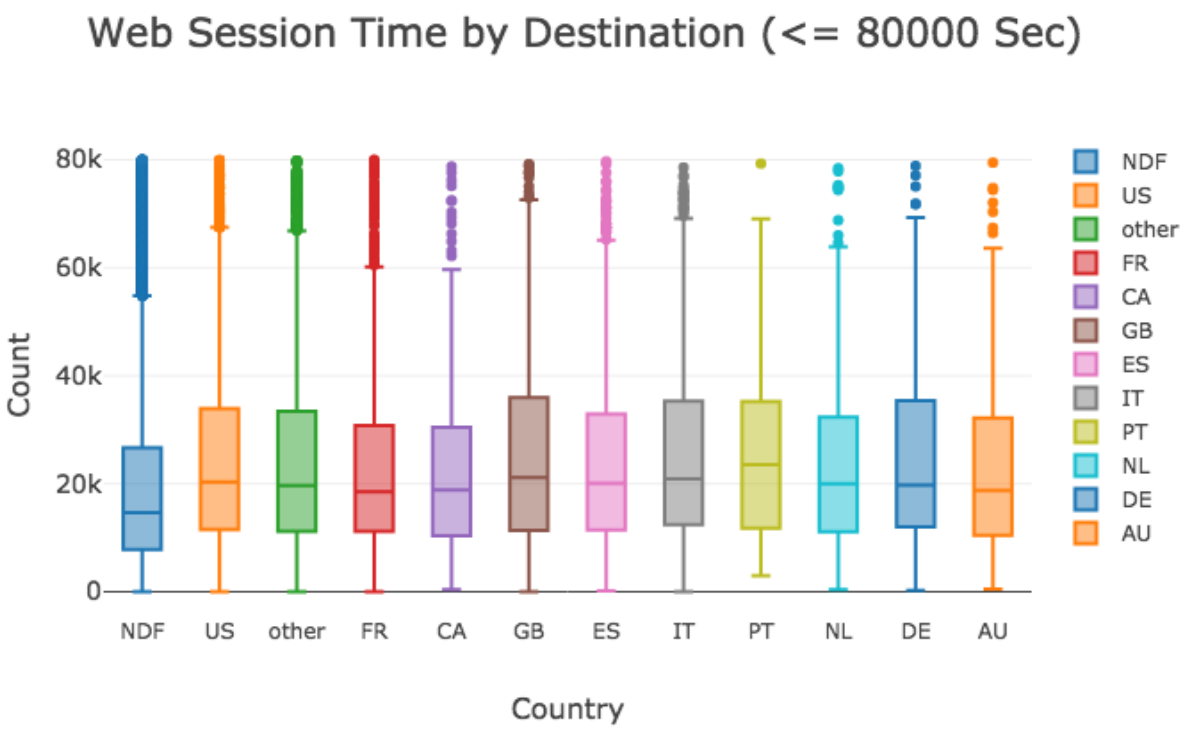

Distribution of Age
Probability
0.06
0.05
0.04
0.03
0.02
0.01
0.00
20
40
60
80
100
age


Distribution of Destination by Age
<= 40y.o.
> 40y.o.
Probability
0.4
0.3
0.2
0.1
0
NDF
US
other
FR
IT
ES
GB
CA
DE
NL
AU
PT
Country

- The majority of Airbnb users are under 40. Younger users (<= 40 years old) had a higher probability of booking a trip (i.e. lower NDF) than those who are > 40 years old. A one-tail z test is conducted to test for it.
  Null hypothesis: Probability of younger users (<= 40 years old) booking a trip - Probability of older users (> 40 years old) booking a trip = 0
  Alternative hypothesis: Probability of younger users (<= 40 years old) booking a trip - Probability of older users (> 40 years old) booking a trip > 0
  The p-value of our test is 7.20e-68. Since the p-value is very close to 0, we reject the null hypothesis and conclude that younger users had a higher probability of booking a trip.
- Younger users were also more likely to pick U.S. as the destination of their first trip. Again, a one-tail z test is conducted to test for the below null & alternative hypothesis
  Null hypothesis: Probability of younger users (<= 40 years old) booking a trip to US - Probability of older users (> 40 years old) booking a trip to US = 0
  Alternative hypothesis: Probability of younger users (<= 40 years old) booking a trip to US - Probability of older users (> 40 years old) booking a trip to US > 0
  The p-value of our test is 3.45e-66. We reject the null hypothesis because the p-value is very close to 0 and conclude that younger users had a higher probability of booking a trip to the U.S.


No. of Web Session by Destination
2500
2000
1500
1000
500
0
Count
NDF
US
other
FR
CA
GB
ES
IT
PT
NL
DE
AU
Country
NDF
US
other
FR
CA
GB
ES
IT
PT
NL
DE
AU


No. of Web Session by Destination (<= 300 Sessions)
300
250
200
150
100
50
0
Count
NDF
US
other
FR
CA
GB
ES
IT
PT
NL
DE
AU
Country
NDF
US
other
FR
CA
GB
ES
IT
PT
NL
DE
AU



- Regarding the number and length of web sessions, seems like those who didn't book a trip had less and shorter sessions.

Those who booked a trip had more web sessions

- Null hypothesis: Mean number of web sessions among those who booked a trip - Mean number of web sessions among those who did not book a trip = 0
- Alternative hypothesis: Mean number of web sessions among those who booked a trip - Mean number of web sessions among those who did not book a trip > 0
- A one-tail z test is conducted to test for the mean no. of sessions between those who booked a trip vs those who didn't. Z test is used because our sample sizes are larger than 30. It is a one-tailed test because we are interested in whether the one mean is greater than the other.
- The p-value of our test is 5.12e-104. Since the p-value is very close to 0, we reject the null hypothesis and conclude that those who booked a trip visited Airbnb's website/ app more frequently.

Those who booked a trip had longer web sessions

- Null hypothesis: Mean web session length among those who booked a trip - Mean web session length among those who did not book a trip = 0
- Alternative hypothesis: Mean web session length among those who booked a trip - Mean web session length among those who did not book a trip > 0

- Again, a one-tail z test is conducted to test for the mean session length between those who booked a trip vs those who didn't because our sample sizes are larger than 30 and we are interested in whether the one mean is greater than the other.
- The p-value of a one-tail z test is 4.68e-34. We reject the null hypothesis and conclude that those who booked a trip had longer web sessions.

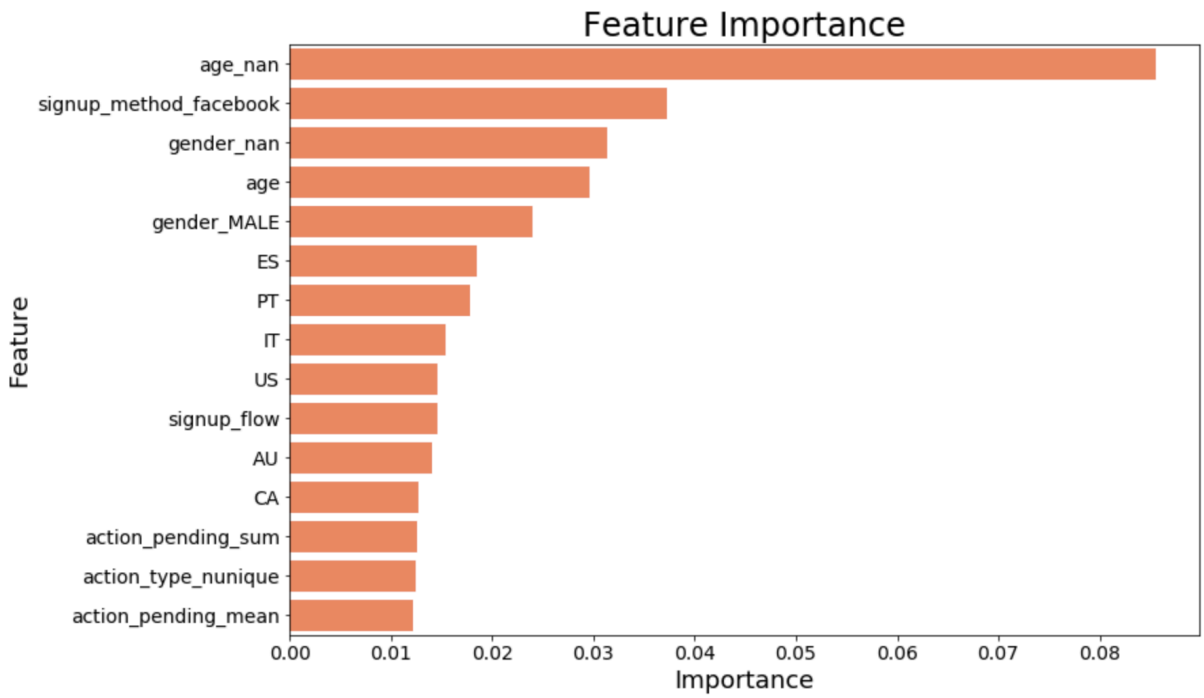
Machine Learning:

A multi-class classification is implemented using users' destinations as the outcome variables and other variables related to users' demographic data, web session records, and some summary statistics of different countries as features. For each user, the top 5 countries with the highest predicted probability are chosen as the predicted output. Predictions based on the test data are submitted to Kaggle and are evaluated using NDCG (Normalized discounted cumulative gain).

Classification Models:

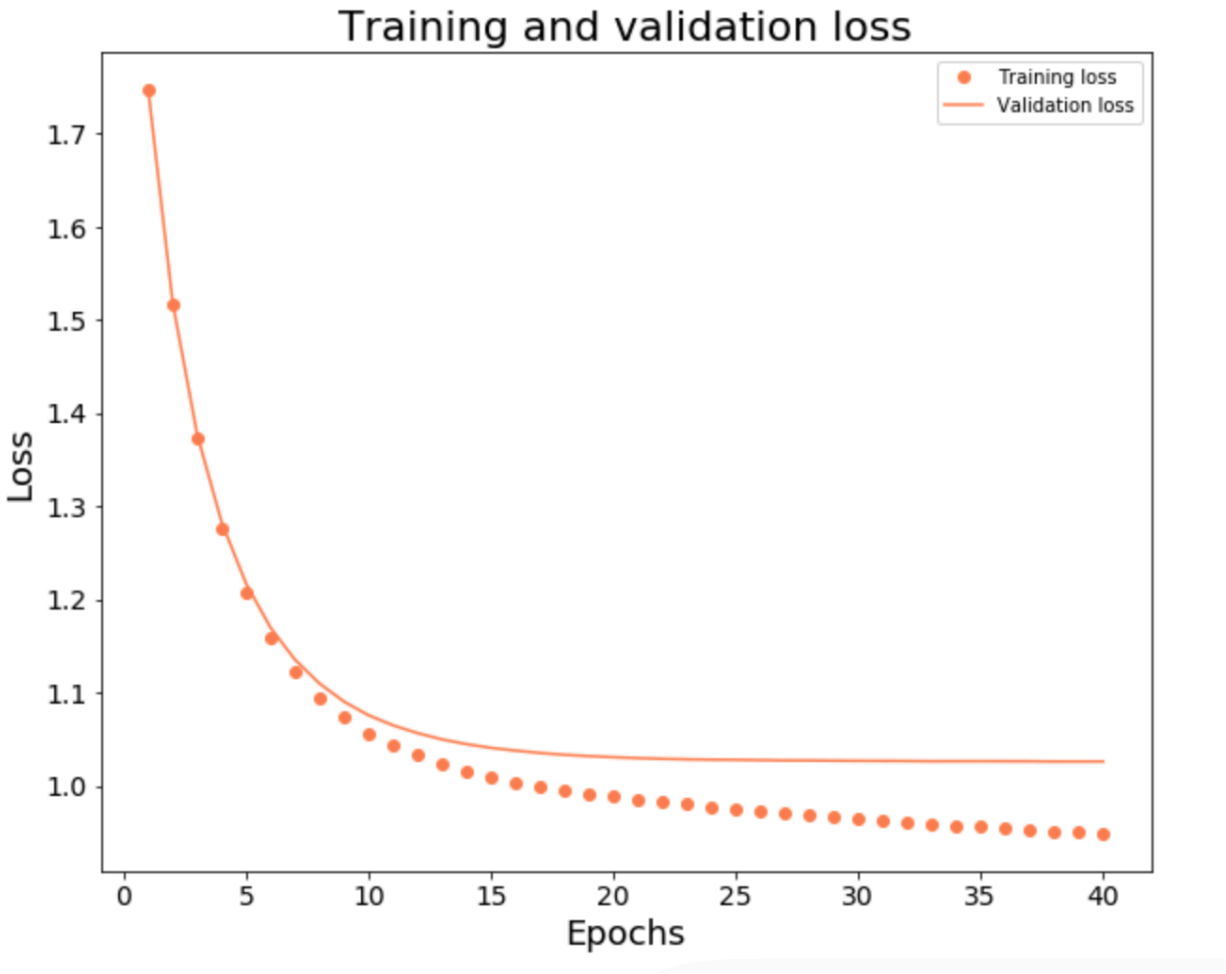
Random Forest with n_estimator=200

- Random forest is used because it can reduce overfitting by introducing randomness through bagging with different combinations of features.
- Grid search with K-Fold cross validation (k=3) is implemented to determine the best parameters for max_features (between 'auto' and 'log2') and min_samples_split (ranging from 0.0001 to 0.05).
- The best parameters are max_features= 'auto' and min_samples_split=0.0005 and the best model gives an NDCG of 0.87521.
- Looking at the feature importance of the best random forest model, the most important feature for predicting destination is whether the user leaves 'age' blank, which makes sense because those who did not intend to book a trip would probably leave 'age' blank. Other important features are whether the user signs up through Facebook, gender and age.



XGBoost

- I try XGBoost because it outperforms other algorithms and is usually one of the winning solutions in Kaggle competitions.
- XGBoost with number of rounds = [10, 20, 40] are run.
- Training and validation loss is monitored, and the training process will stop if the validation loss has not improved for 5 rounds to avoid overfitting.
- The model with number of round = 40 gives the best NDCG of 0.87837 although the validation performance doesn't seem to improve much after 15 rounds. Early stopping didn't kick in because the validation loss was still decreasing slightly (although it is hardly noticeable).



Feedforward Neural Networks

- Neural Networks are considered as they are flexible - by varying the number of layers and nodes, neural networks can fit different data with different complexity.
- Feedforward Neural Networks with varying depths (from 1 to 5) and number of nodes (from 50 - 400) are tried. Features are normalized before feeding into the Neural Networks.
- Again, early stopping is implemented by monitoring the training and validation loss.
- Batch size = 50 and no. of epoch = 20 are used, and the model hits early stopping after 10 epochs.
- The best model is the one with 4 layers and number of nodes = [250, 125, 125, 12]. It gives an NDCG of 0.87414.

Ensemble Model (Soft Voting)

- Weighted mean of the predicted probabilities from the best Random Forest, XGBoost and Neural Network model are calculated. For each user, the top 5 countries with the highest weighted mean are chosen as the predicted output.
- Different combinations of weights are tried and the combination that gives the best model is [Random Forest, XGBoost, Neural Networks] = [1, 2, 1]. It gives an NDCG of 0.8780.

Final Model:

My final model is XGBoost Model with 40 rounds of training with a NDCG score of 0.87837. It ranks 325 out of 1462 on Kaggle Leaderboard.

Future Research Recommendation:

- More hyperparameter tuning: If given more time and resources, I will try to tune more hyperparameters to optimize my models.
- Try more classification algorithms: If given more time and resources, I will also try more algorithms such as AdaBoost, CATBoost, etc.