AirBnB Recruiting Challenge

Springboard DATA SCIENCE INTENSIVE - capstone project

Ruben BLOOM

2016

# Acknowledgement

I give thanks and credit to my Springboard mentor, Karthik Ramasay**,** who advised me throughout this project.

# Summary

Technical section is detailed.

4. Data Preparation begins my technical work.

**Note:**This is a formal report which is likely overkill for every day communication of results. Readers may prefer a single Jupyter/iPython notebook which presents the technical work and findings more succinctly. The Jupyter is closer to what I expect to share internally in a company. See here:

Contents

[Acknowledgement 1](#_Toc440110903)

[Summary 1](#_Toc440110904)

[Introduction 3](#_Toc440110905)

[Problem Definition and the Data 4](#_Toc440110906)

[Imbalance/Skew 4](#_Toc440110907)

[Scoring 5](#_Toc440110908)

[Data Quality 5](#_Toc440110909)

[Data Preparation 6](#_Toc440110910)

[Wrangling and Cleaning 6](#_Toc440110911)

[Feature Extraction and Selection 6](#_Toc440110912)

[Machine Learning Models 6](#_Toc440110913)

[Approach 6](#_Toc440110914)

[Linear Regression 6](#_Toc440110915)

[Random Forest 6](#_Toc440110916)

[Class Imbalance (Skew) 7](#_Toc440110917)

[One-vs-One Classifier 7](#_Toc440110918)

[Ensemble of Classifiers 7](#_Toc440110919)

[Results 7](#_Toc440110920)

[NDCG 7](#_Toc440110921)

[Confusion Matrices 7](#_Toc440110922)

[Feature Importances & Coefficients 7](#_Toc440110923)

[Validation Curve 7](#_Toc440110924)

[Further Ideas 7](#_Toc440110925)

[Modelling Conclusion 8](#_Toc440110926)

[Practical Recommendations 8](#_Toc440110927)

# Introduction

The project described in this report is primarily a submission to the AirBnB Recruiting New User competition on Kaggle.com[[1]](#footnote-1), but with the important addition of recommendations based on my findings. This report presents my technical approach and my discoveries about the problem.

The challenge is best introduced with AirBnB’s own description:

New users on Airbnb can book a place to stay in 34,000+ cities across 190+ countries. By accurately predicting where a new user will book their first travel experience, Airbnb can share more personalized content with their community, decrease the average time to first booking, and better forecast demand.

In this recruiting competition, Airbnb challenges you to predict in which country a new user will make his or her first booking.

As an example of content personalisation, when users land on the AirBnB homepage or open the AirBnB app, they are presented with suggested destinations based on predictions about where they are likely to book. Don’t show Paris and Rome to users who are mostly likely looking for a weekend getaway near home.

For the Kaggle competition, AirBnB has made available datasets of new users’ information and their ultimate destinations (a labelled dataset) to allow for the creation of predictive machine learning model created with supervised learning algorithms.

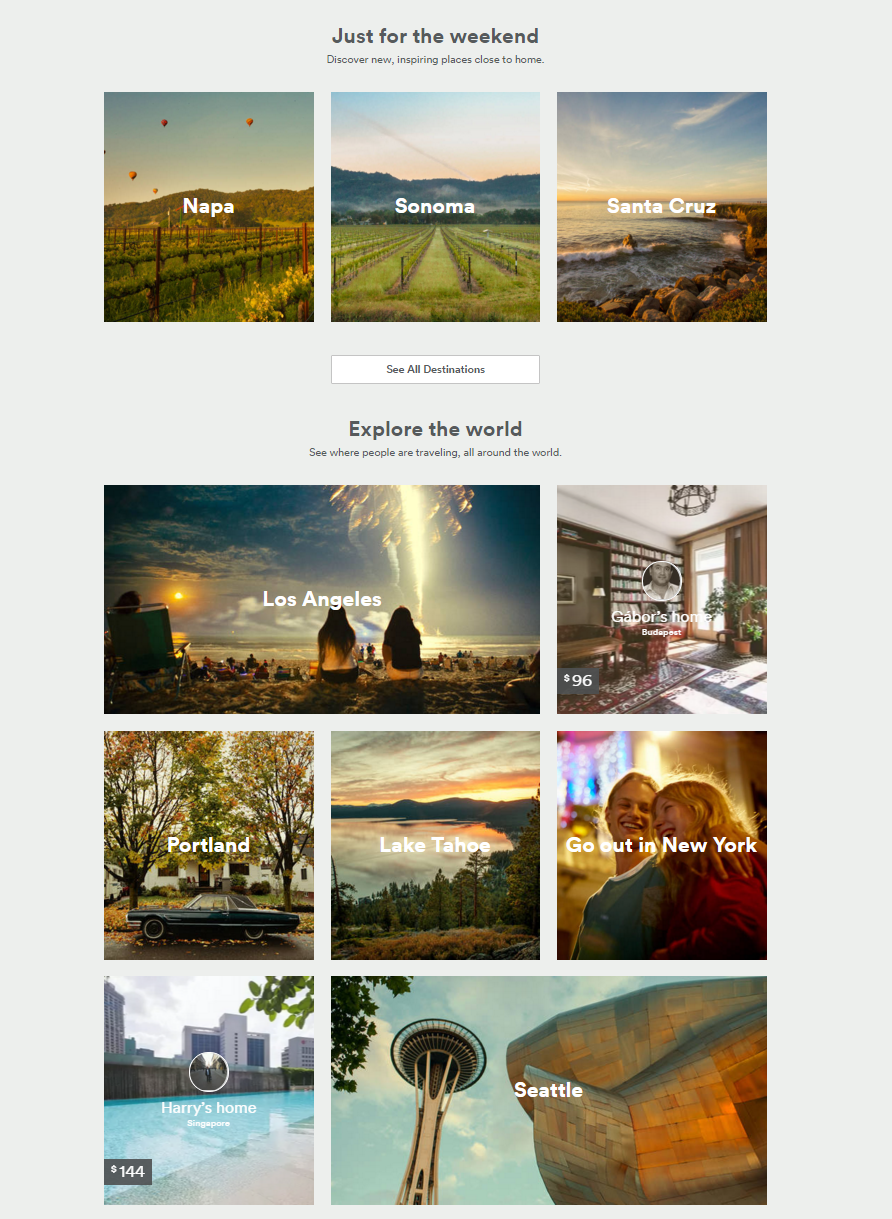
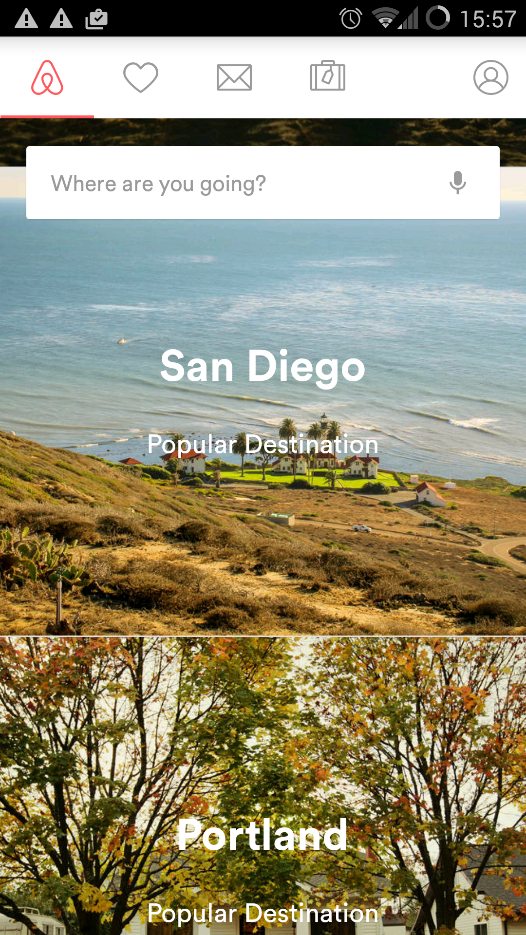
 

Figure Left: AirBnB.com Home page displaying suggested travel destinations. Right: Suggested destinations on AirBNB mobile app.

# Problem Definition and the Data

AirBnB have made available several files for the competition. Two consist of specific records for users (three if you include the test set), and another including general demographic information and information about the countries.

**train\_user.csv / test\_users.csv**This is the primary database and includes user age, gender, language, time first active, date account created, sign up method, sign up flow, affiliate information, first device, first browser, and ultimate country destination of first booking.

There are 213,451 users in the training set and 62,096 in the test set. Training Users have first activities in the period 2010-2014. Test users are new users starting from 7/1/2014 (July).

Although AirBnB has destinations in more than 190 countries, only 12 destinations are included in this dataset: USA, Canada, Great Britain, Australia, France, Italy, Germany, Spain, Portugal, Netherlands, “other” (not an above listed country), or NDF/ “No Destination Found”, i.e. no booking was made.

All users are from the US.

## Imbalance/Skew

The dataset is extremely imbalanced, with the US and NDF being supermajorities with tens of thousands of examples, and all other destinations having a few thousand or few hundred, see Figure 2. This makes training a machine learning model to predict minority cases difficult.

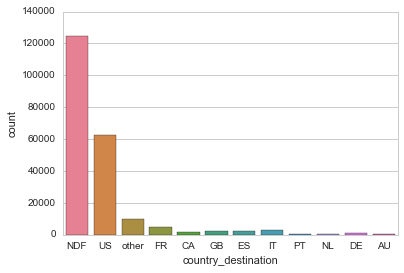


Figure Histogram of Users travelling to each destination.

**sessions.csv**The sessions file contains browsing information for users at the level of individual actions. Action, action type, action detail, device type (Windows/Mac/Android/Mobile) and seconds elapsed since previous action are included. Examples rows in Table 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| user\_id | action | action\_type | action\_detail | device\_type | secs\_elapsed |
| d1mm9tcy42 | personalize | data | wishlist\_content\_update | Windows Desktop | 1399 |
| d1mm9tcy42 | index | view | view\_search\_results | Windows Desktop | 74886 |

*Table 1: Examples of sessions data.*

The sessions data is only available for users first active after the start of 2014.

**age\_gender\_bkts.csv**This file contains the number of users in each age and gender bracket, e.g. (Female, 24-29), travelling to each destination in 2015.

**countries.csv**This file contains information on each country destination, including latitude and longitude of the country, distance from the US, size of the country, language in the country, and the Levenshtein distance[[2]](#footnote-2) from English to the country’s language.

Scoring

The goal is correctly predict the country destination for each user, however this is not scored using simple accuracy of wrong or right. Instead, the competition is scored using Normalised Discounted Cumulative Gain (NDCG)[[3]](#footnote-3). Each prediction should be a list of 5 destinations which are mostly likely for the user to have booked in, in order of likelihood. The score is highest when the actual destination is at the top of the list, second highest when correct destination is second in the list, and so on.

## Data Quality

It should be noted that the quality of the data is unlikely to perfect. As with most user data collected online, the recording is imperfect.

Consider that AirBnB attempts to log when a user was first active, as well as track their sessions. This is difficult because of the Many Screens Problem[[4]](#footnote-4). A user may have been browsing extensively on one computer, perhaps at work, and then make the actual booking at home. It will appear that the user booked spontaneously without browsing. Similarly, many people browse on mobile but book at full computer. Until a user has created an account and is signed in, this is difficult to track.

Ultimately, we cannot be sure that any recorded data about a user is the complete picture of their behavioural pattern. We should expect that the data is noisy.

# Data Preparation

### Wrangling and Cleaning

42% of age values and 47% of gender values were missing in the dataset. Many values were unlikely, being less than 15 or greater than 100. These invalid values were set first set to Missing (NaN).

I differentially cleaned the data depending on the type of model to be used. For a logistic regression model, I set the missing age values to the median age, and additionally scaled the values to fall between 0 and 1. For a Decision Tree/Random Forest model which can handle non-linear data, I set missing ages to -1. The rational for this approach is that setting all missing values to distinct number allows the Decision Tree to treat them as their own class. This is valuable, because while imputation to a likely age value might help predict booking destination based on demographic information, there may be significant differences in booking behaviour between 24 year olds who filled in their age and those who didn’t. The latter is less likely to book at all.

Additionally, for both Logistic Regression and Decision Tree models, I created discretised age features back on brackets, e.g. [20-24],[25-29],[30-34]. These features are 0/1 is the user isn’t/is in the bracket.

Similarly, all categorical variables were converted to dummy variables/one-hot encoding. E.g. First Device Type feature with the options of “Windows”, ”MacDesktop”, ”iPhone” is converted to DeviceType\_Windows, DeviceType\_MacDesktop. 1/0 is placed in a feature appropriately.

One-hot encoding was applied to gender as well. Users who did not fill out their gender would be 0 in all gender features (gender\_FEMALE, gender\_MALE, gender\_OTHER).

Time First Active and Date Account Created were broken down into Year, Month, Day, Hour features to allow the models to pick up patterns at those time scales.

### Feature Extraction and Selection

––

Processing of Sessions  
- Aggregation of Action Types  
- Use of elapsed seconds to get length of sessions, number of sessions, period active.

Use SelectKBest to reduce. Generally rely on regularisation, lowering number reduces computational cost.

Age Gender Brackets  
- Normalise. Add to each individual. Add overall prior where information is missing.

# Machine Learning Models

## Approach

Use 5 most frequent classes. Only records with session data (most recent)

## Linear Regression

Simple, Interpretable

Tuning by GridSearchCV

Bad: One-vs-rest needed. Many classes! Slows down iteration.

## Random Forest

Inherently multiclass. Handle non-linearities well. Ensemble method. Resistant to overfitting. (Many, many features!)

## Class Imbalance (Skew)

Class Weights (no difference)  
Oversampling  
Undersampling  
Undersamlping + SMOTE (synthetic minority oversampling)

## One-vs-One Classifier

Small classes vs. small classes. No improvement.

## Ensemble of Classifiers

NDF vs Rest, Destinations vs Each Other  
NDF vs Rest did not have super high accuracy.  
Rest amongst selves was worse than dummy estimator.

# Results

Best model, LB of 0.87287. Comparison. CV. 109th out 835. 1st place is at 0.88107. Dummy Estimator at 0.85 . . . Random Forest, specifications . . .

### NDCG

### Confusion Matrices

Always select the majority class.  
Classification Report

## Feature Importances & Coefficients

Mobile Devices Convert Less  
Mac Desktop users do so more often  
Certain sign-up flows convert more.

### Validation Curve

Increasing complexity doesn’t help

## Further Ideas

Try including other locations, maybe easy to identify. Seems unlikely.  
Use countries file . . .  
Two classifiers – one without session data, one with. Not done because session data is much richer.

# Modelling Conclusion

What – dataset is insufficiently rich to allow for high accuracy.

Why I reached this conclusion:

Observations that: Validation and Learning Curves, Model Complexity, Inability to Overfit (Random Forest out-of-bag), No increase with more train samples. Insufficient features – failed to engineer, or info just isn’t there.

Reasons why it is logical to assume it isn’t there.  
Data Quality – multiple screens and tracking problem. Maybe searched elsewhere. Maybe searched extensively and then booked elsewhere after signing in.  
Browsing behaviour is not significant differentiated by these features for different destinations. E.g. Spain and Great Britain visitors. (Should try fclassif without NDF?)

# Practical Recommendations

Find data with richer features.  
If restricted to such a dataset, focus on predicting booking vs. not booking. Offer incentives to non-bookers to book (e.g. discount codes)

1. https://www.kaggle.com/c/airbnb-recruiting-new-user-bookings [↑](#footnote-ref-1)
2. https://en.wikipedia.org/wiki/Levenshtein\_distance [↑](#footnote-ref-2)
3. https://en.wikipedia.org/wiki/Discounted\_cumulative\_gain#Normalized\_DCG [↑](#footnote-ref-3)
4. http://www.kaushik.net/avinash/multi-channel-attribution-definitions-models/ [↑](#footnote-ref-4)