**Data Science Project Protocol  
Naama Jan**

**Introduction:**

For this project, I decided to choose the Airbnb Berlin dataset.  The dataset can be found on Kaggle:  
<https://www.kaggle.com/brittabettendorf/berlin-airbnb-data>

[Airbnb](https://www.airbnb.com/) is an online marketplace where people can offer and rent out their houses and apartments (properties). Airbnb became extremely successful for tourists and business people, who are looking for an overnight accommodation.

The datasets are based on properties list with description (features) and information whether the properties are being rented in each day from November 2018 until November 2019.

The project goal is predicting the probability of the properties to be rented in the next 3 months. Defining the outcome variable as booked if there are at least 70 rented days in the 3 months following every month.

The outcome variable is the "booked up" state of the following 3 months for each month starting in March until August (the first prediction starts in March since I'm using November-February for training. August is the last month since the last prediction is from August until November). I define the booked-up variable as a binary variable where it's set to 1 if at least 70 out of 90 days are booked and 0 otherwise.

I plan to examine the occupancy of the properties, according to the variables in the model. In general, there are variables that I expect to have high significant influence on the outcome. For example, price, weather, location (neighborhood), the distance to the city center, the size of the property, room type, reviews (especially review scores rating) and etc.

**Data**

The main datasets are based on properties list with description (features) and information whether a property is being rented in each day from November 2018 until November 2019 (every record describes property in a day).

As mentioned in the introduction, our target is predicting whether a property will be rented in the next 3 months (defining the booked\_up\_target base on occupancy of the property).

These datasets are taken from [Kaggle](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data) and ensembled from 6 different csv files:

* [Calendar Summary](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=calendar_summary.csv)
* [Listings Detailed](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=listings.csv)
* [Listings Summary](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=listings_summary.csv)
* [Neighborhoods](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=neighbourhoods.csv)
* [Reviews Detailed](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=reviews.csv)
* [Reviews Summary](https://www.kaggle.com/brittabettendorf/berlin-airbnb-data?select=reviews_summary.csv)

For enrichening the main data sources, I used [climatestotravel](https://www.climatestotravel.com/climate/germany/berlin) website for getting average weather data. Assuming that weather info can influence on tourism in Berlin which can directly influence the demand for properties.

Importing the following datasets from this website:

* Sunshine hours
* Average precipitation
* Average temperatures

The data sections in this project are divided to the following steps:

1. The first step in the project is collecting the datasets from Kaggle to SQL server and creating a flat table (by relevant joins between the tables). Most of the features are created at this stage.
2. There are enrich features that requires analysis of text. These feature have been added to flat table using “1- Addition to Flat file.ipynb” notebook.
3. In “[2 - Berlin\_Airbnb\_EDA.ipynb](https://github.com/NaamaJan/airbnb-berlin-availability/blob/main/2%20-%20Berlin_Airbnb_EDA.ipynb)” notebook, I perform an analysis of the detailed Berlin listings data. I am using visualizing and analyzing data to extract insights from the variables in the data.

In the beginning, I used descriptive statistics to explore the data, which can help describe the data set's basic features and obtain a summary of the data. Then, I perform Data Visualization analysis to provide an accessible way to see and understand trends, outliers, relationships, variability, patterns in data and to notice if there is a problem with data quality.

I have been paying attention to the correlations and the differences between the variables in the data set and describe the target variable "booked\_up\_target", his distribution and its relationships with the variables.

This section also helped me create new variables or perform variable transformations.

1. [3- Berlin\_Airbnb\_Data\_Cleansing.ipynb](https://github.com/NaamaJan/airbnb-berlin-availability/blob/main/3-%20Berlin_Airbnb_Data_Cleansing.ipynb) notebook includes reduce the bias by filling NA, fixing outliers and etc. This is based on the results and conclusions of the EDA section.

The methodology of handling outliers:

1. If the outliers are a mistaken observation - replacing the outliers with na.
2. If the outliers generate any false correlation with the outcome - replacing the outliers with na.
3. If the removal of the outliers changes the distribution of the outcome but not the correlation - replacing the outliers with na.
4. If the removal of the outliers changes the both correlation and the distribution of the outcome - we can't replace the outliers with na. In this case we will need to perform a data transformation or make the variable categorical and thus divide it into groups that one of which will be "missing" or use models to predict the missing variables.

The methodology of handling missing values depends on the mechanism of missing generation for each variable. For each variable we need to decide the correct method to be applied:

1. Dropping rows with more than 60% outliers.
2. checking the number of NA in columns:
   1. If the percentages of na are greater than 70% -> dropping the column.
   2. If the percentages of na between 40%-70% -> transforming the variable categorical.
   3. If the percentages of na are lower than 40% -> depends on the mechanism of missing generation for each variable. For each variable, we have to decide the correct method to be applied. If the mechanism of missing is MNAR -> There is an explanation of why the value is missing. In this case, we can transform the variable categorical or drop the column, we cannot do imputation. Otherwise, if the missing mechanism is MCAR or MAR, we can use imputation techniques.
3. [4- Feature Enrichment.ipynb](http://localhost:8888/notebooks/Documents/projects/airbnb-berlin-availability/4-%20Feature%20Enrichment.ipynb) notebook includes 3 way of adding features:
   1. Feature Extraction: obtaining new features from existing features.
   2. Feature Engineering: transformation of raw data into features suitable for modeling.
   3. Feature Transformation: transformation of data to improve the accuracy of the algorithm.

## For creating [Data retrieval protocol](https://github.com/NaamaJan/airbnb-berlin-availability/blob/main/Data%20Retrieval%20Protocol.xlsx) I used [7 - Data Retrieval Protocol Helper.ipynb](https://github.com/NaamaJan/airbnb-berlin-availability/blob/main/7%20-%20Data%20Retrieval%20Protocol%20Helper.ipynb) **notebook.**

**Models**

In the model section I am planning to divide the data to train, dev and test datasets partitions.

The datasets should be spit only after shuffle and balance between the 3 datasets.

pyMechkar package include utils for splitting the data, shuffle and supply indication whether the data is balanced (tuning with seed and prop parameters). The flat table includes ~157K records, so it seems reasonable splitting having propitiation of ~60%, ~20% and 20%. I will use this package.

The first stage of this splitting is defining 20% of the records as test partition, keeping it aside and not using it until the end of the project.

The 80% leftovers should be split to train and dev with proportion of 80%-20%. The train dataset will be used to train the models while the development dataset will be used for assessment of the model performance.

The outcome 'booked\_up\_target' is binary variable which is set to 1 if at least 70 out of 90 days are booked and 0 otherwise. For training the model I will use wide range of classifier models with default params, selecting the model which perform best on the train and dev partition (high score and balance scores between 2 partitions).

The score will be determined by Area Under the Curve (AUC) metric, which can be useful in the scenario which the 72% of that target are “1” and 28% are “0”.

Once selecting the best model (base on train and dev), I will make fine-tune of the hyper-paremeters:

1. Depending on the model, create vectors with a wide range of different values for part of the parameters that may affect the performance of the model.
2. Using Grid-Search for finding the best matching parameters.