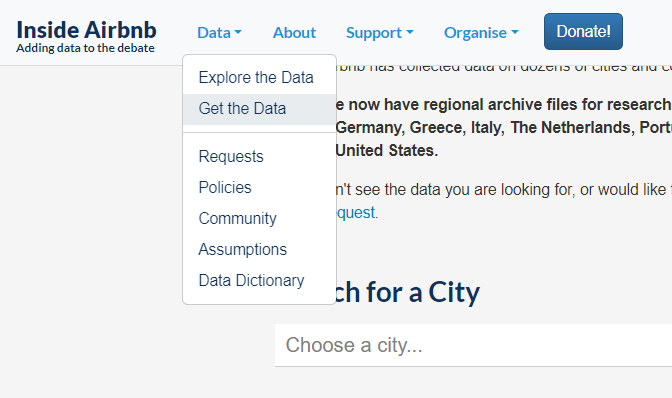
## Data Analysis Term Project

Your task is to analyze Airbnb data using SAS. The below synopsis lists the tasks to be completed. The

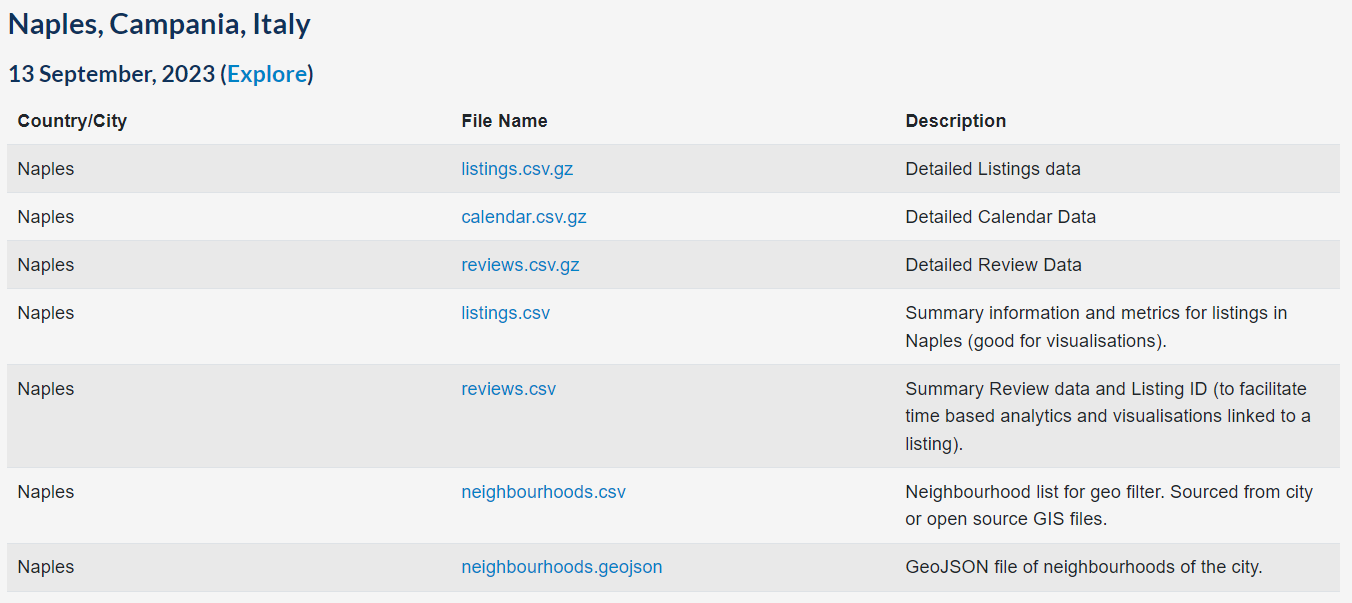
## Data

The data for the project is available at <http://insideairbnb.com/explore/>. This website includes sample Airbnb data from selected locations around the world. Each student is to select one city to focus their analysis on. Each student is to pick a different city and notify the instructor on a first-come-first-served basis. The instructor will have the selection list available in D2L for reference.

Once a city is picked, click on the **Data** tab at the top of the page.



Each city will have the same set of files available for download. For instance, files available for Naples, Italy are listed in the listing below. For the purpose of this exercise, you will need to download the **listings.csv.gz** file. GZIP archives can be extracted with software like 7-Zip (<https://www.7-zip.org/>) on Windows or Keka on Mac (free download available at <https://www.keka.io/en/>)



### Data Dictionary

(adapted from <https://docs.google.com/spreadsheets/d/1iWCNJcSutYqpULSQHlNyGInUvHg2BoUGoNRIGa6Szc4/edit?usp=sharing>)

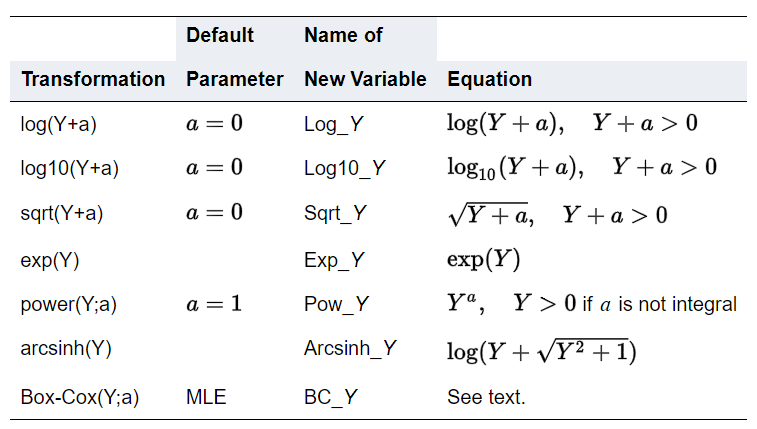
The file you download will have several other columns included – below, I listed only those variables that might be useful for fitting a regression model.

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| id | integer | Airbnb's unique identifier for the listing |
| name | text | Name of the listing |
| description | text | Detailed description of the listing |
| neighborhood\_overview | text | Host's description of the neighborhood |
| host\_response\_time |  |  |
| host\_response\_rate |  |  |
| host\_acceptance\_rate |  | That rate at which a host accepts booking requests. |
| neighbourhood\_cleansed | text | The neighborhood as geocoded using the latitude and longitude against neighborhoods as defined by open or public digital shapefiles. |
| latitude | numeric | Uses the World Geodetic System (WGS84) projection for latitude and longitude. |
| longitude | numeric | Uses the World Geodetic System (WGS84) projection for latitude and longitude. |
| property\_type | text | Self selected property type. Hotels and Bed and Breakfasts are described as such by their hosts in this field |
| room\_type | text | [Entire home/apt|Private room|Shared room|Hotel]  All homes are grouped into the following three room types: Entire place Private room Shared room |
| accommodates | integer | The maximum capacity of the listing |
| bathrooms\_text | string | The number of bathrooms in the listing.  On the Airbnb website, the bathrooms field has evolved from a number to a textual description. For older scrapes, bathrooms is used. |
| bedrooms | integer | The number of bedrooms |
| beds | integer | The number of beds |
| price | currency | daily price in local currency |
| minimum\_nights | integer | minimum number of night stay for the listing (calendar rules may be different) |
| maximum\_nights | integer | maximum number of night stay for the listing (calendar rules may be different) |
| has\_availability | boolean | [t=true; f=false] |
| availability\_30 | integer | avaliability\_x. The availability of the listing x days in the future as determined by the calendar. Note a listing may not be available because it has been booked by a guest or blocked by the host. |
| availability\_60 | integer | avaliability\_x. The availability of the listing x days in the future as determined by the calendar. Note a listing may not be available because it has been booked by a guest or blocked by the host. |
| availability\_90 | integer | avaliability\_x. The availability of the listing x days in the future as determined by the calendar. Note a listing may not be available because it has been booked by a guest or blocked by the host. |
| availability\_365 | integer | avaliability\_x. The availability of the listing x days in the future as determined by the calendar. Note a listing may not be available because it has been booked by a guest or blocked by the host. |
| number\_of\_reviews | integer | The number of reviews the listing has |
| number\_of\_reviews\_ltm | integer | The number of reviews the listing has (in the last 12 months) |
| number\_of\_reviews\_l30d | integer | The number of reviews the listing has (in the last 30 days) |
| review\_scores\_rating |  |  |
| review\_scores\_accuracy |  |  |
| review\_scores\_cleanliness |  |  |
| review\_scores\_checkin |  |  |
| review\_scores\_communication |  |  |
| review\_scores\_location |  |  |
| review\_scores\_value |  |  |
| instant\_bookable | boolean | [t=true; f=false]. Whether the guest can automatically book the listing without the host requiring to accept their booking request. An indicator of a commercial listing. |
| reviews\_per\_month | numeric | The number of reviews the listing has over the lifetime of the listing |

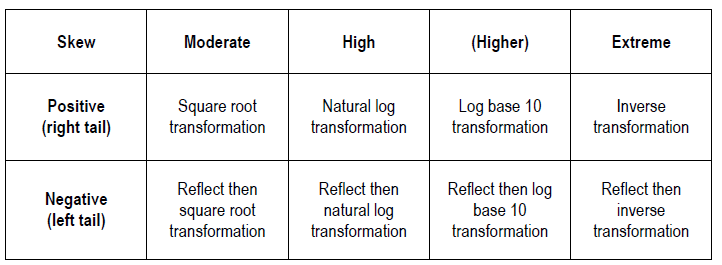
## Data Analysis

### Linear Regression

1. Import the data into SAS (use PROC IMPORT)
2. Fit a linear regression model using price as the dependent variable.
   1. Investigate the dependent variable:
      1. Are there any missing or nonsensical values? Use PROC UNIVARIATE, which will provide you with information about the distribution and range of a continuous variable (Program 2.6 on p. 49 can help you with the setup)
      2. If you identified any outliers, include the corresponding output and consider removing them from the dataset (see Program 2.7 on 52 for further information)
      3. PROC UNIVARIATE will also provide you with information about the shape of the distribution of the PRICE variable. Ideally, we would like it to be normally distributed. Different transformations are available for significant departures from normality. Our book uses the log transformation, but other common options include:



* + 1. If you find the data to be skewed, below are some common guidelines on which transformation to choose for a given skewness level. Program 2.7 includes an example of a log transformation in action.



* 1. Investigate the predictors:
     1. Numeric variables:
        1. Is predictor multicollinearity an issue? If so, make sure that you address the issue.
           1. Program 2.9 includes an interesting approach to creating a correlation table.
           2. You can also fit a regression model, include Variance Inflation Factors in the output, and check for multicollinearity (see Program 2.10 for more details).
        2. Categorization – are there any categorical variables with many levels (for instance, more than 5-6)? If so, consider collapsing them into smaller number of categories. You will need PROC FREQ for the initial evaluation. Program 2.17 includes a sample code demonstrating how to collapse categorical levels. Remember that when you collapse (i.e., bin) numeric variables into categorical variables, the main goal is to have similar or comparable counts (or at least in the same order of magnitude) for each category. Chapter 5 of *End-to-End…* text contains a discussion on this topic.
        3. Missing values – do any continuous variables include missing values? If so, you can keep them as they are (consequently, SAS would exclude them from the analysis) or use a missing value imputation technique (e.g., an average of the other values). Missing value imputation should only be considered if the number of missing values is relatively small (say, <5% of observations have a missing value on a given variable).
        4. You should also consider feature engineering (i.e., creating higher-order terms or polynomials) and standardizing numeric variables to bring variables to the same scale. Chapter 5 of *End-to-End…* text contains a discussion on these topics. In general, we can plot each predictor against the dependent variable to try and assess what type of relationship might exist between the two variables.
     2. Character (categorical variables).
        1. Create frequency tables to examine the distribution of the categorical variables. You can use PROC FREQ for this purpose.
        2. If variables have too many levels, consider collapsing them into fewer groups.
  2. Split data into train and test samples. You can use an 80/20 ratio (or a different ratio of your choice as long it conforms to generally accepted standards). Programs 2.4 and 2.5 demonstrate how this can be accomplished in SAS. Any data adjustments should be performed before the split so that you do not need to repeat them separately for training and testing data.
  3. Fit a regression model using PROC GLMSELECT and LASSO selection. Use Price as the dependent variable (or the transformed price value if you created one). Programs 2.19 and 2.20 provide a sample syntax for the step.
     1. Examine the regression assumptions.
     2. Interpret the output. Is the model any good? What is the value of RMSE (Root Mean Square Error)? Assess RMSE for both training and testing samples.
  4. Create a decision tree using the PROC HPSPLIT procedure with price (or its transformed version) as the model’s dependent variable. Programs 2.23 and 2.24 contain code samples to help set up the procedure. Use the train data set to fit the model.
     1. Use the scoring code produced by the PROC HPSPLIT procedure to score the hold-out test sample (see program 2.25 for details).
     2. Assess RMSE and compare it against the statistic obtained with the linear regression model. Assess RMSE for both training and testing samples.
  5. Fit a random forest model using the PROC HPFOREST procedure using price (or its transformed version) as the target variable (see program 2.26 for details).

1. Compare the models and state your recommendations.