NYCVA
A visual analysis of New York City AirBnBs

Leonardo Balzoni - 1870364
Alberto Contorno - 1873252

February 21, 2020

1 Introduction to the analysis

Housing websites like *AirBnB* [1] are amongst the sites that globally generate more traffic [2]. Even with the impressive unseen work that ensures the navigating the site is as pleasurable as possible, with the huge amount of data it can be quite hard to get a grasp on the actual informations that we, as users, are looking for. Our work will try to give some, otherwise, hard to see insights about the announcements, with the help of visual analytics. Obviously doing it on a global scale would be too computationally heavy for our resources so the analysis will be limited to one of the cities that has the most announcements on the site; *New York City.*

2 Description of the dataset

The dataset that is going to be used was provided from *Kaggle* [3], and it is made of around 50000 tuples with 16 attributes each [4]. We are perfectly aware of the fact that this means that our *"AS index"* is quite higher than the suggested one, which means that we are faced with 2 options: reduce the amount of attributes and/or tuples, or build a small backend server (probably in *NodeJS*) on which to decouple the computational load to make the web browser more responsive.

Each tuple of the set represents an announcement on the AirBnB website for a house in New York City, some of the most useful attributes are: the location of the house (both its neighbourhood and its coordinates), its cost, the average availability, and the amount of reviews. These information will be used to identify patterns and gather insights about the distribution of rental housing in NYC in order to have a visual representation of the best options available.

3 Goals of the Analysis

An analysis of this kind can be used to fulfill multiple objectives; firstly as already stated the multiple visualizations will allow to see otherwise hidden information, such as for example the average price in each neighbourhood, or the estimated income of the various hosts. This means that both someone who is willing to put a announcement on the website and someone who is looking for an accomodation, will be able to find useful information.

Moreover we could apply some dimensionality reduction algorithms such as *PCA* or *t-SNE* to gather informations about similar announcements, which

could for example be used to find an announcement similar to one that we like, but that might be unavailable.
As for non-functional requirements, we want to make sure that the tool can be used on a wide range of systems and devices, this means that responsiveness and performance will be always taken into consideration during our development.

4 Dataset preprocessing

The given dataset was immediately almost ready to be used, as there were no duplicates and the attribute's types were coherenent and almost always not null.
Even with this said some preprocessing was necessary, we began by dropping some useless columns such as the name of the host or the date of the last review. Moreover there were some announcements with missing data for important fields, so we dropped those (circa 20 out of 50000).
Lastly there was the "Reviews per month" attribute that was null in the case of 0 reviews on the announcement. Instead of dropping these, what we did was to fill the null values with 0 in the case of no reviews for the house.

4.1 PCA preprocessing

While what we said is enough for all the analyses that we ended up doing, evaluating PCA requires for more attention. This is due to the fact that the algorithm only accepts numerical values for the various fields, and our dataset instead has some categorical ones (i.e. "neighbourhood" or "housing_type"). What we did was creating some sort of one-hot encoding for those fields; for example "housing_type" could be: an entire apartment, a private room or a shared room, so we replaced the "housing_type" field with 3 new ones: "housing_type_apt", "housing_type_pvtroom" and "housing_type_shroom", and the values are all 0 except for one which is 1. Doing the same for the rest of the categorical fields we finished the preprocessing for the dataset for PCA.

5 Tools used

The code is mainly written in *Javascript* and *HTML*, although at the core of our analysis is *D3.js* [5], a small, free javascript library for manipulating

documents based on data. Instead of using the basic D3 functions we also utilised *PlotlyJS* [6], a high-level, declarative charting library built on top of D3, which ships with lots of chart types.
It also has to be stated that the preprocessing of the dataset was done in *Python* [7] with the aid of *Pandas* [8], a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.
Lastly to make things like HTML document traversal, manipulation and event handling much simpler we also used *jQuery* [9] .

6 Visualizations

6.1 Average neighbourhood price barchart

sia neigh group che singoli hood (specifica select per aggiungere/levare)

6.2 Amount of announcements per neighbourhood barchart

specifica che hai diviso per roomtypes con stacked barchart

6.3 Violin plot and barchart for pricing distribution across neighbourhoods

ricorda del bottone per switchare fra i 2

6.4 Word Cloud

utile per vedere parole chiave

6.5 Map Scatterplot with encoding for price and average income

specifica preprocessing per calcolare income mensile dell'announcement

6.6 PCA

mortacci sua, ricorda considerazioni su similitudini grazie a z-norm

7 Interaction design

Visualizing data is one thing, being able to interact with it is another. We wanted to make it possible for the user to be able to interact with all the graphs in our project, and the choice of using *PlotlyJS* surely helped in this regard. Thanks to it, all the graphs done using the library are: pannable, zoomable and selectable both with a box selection or a lasso selection tool; moreover the user is also able to reset the graph to its original view and download it as a png image.

Even though this already makes a huge difference with respect to static data, we wanted something more. There are multiple scatterplots across the project (in the pricing distribution graph, on the map, and on the PCA representation) and we thought that it would have been interesting to make the selection in a across-project manner. This means that when the user selects some houses in one of these graph, the selection is carried across to all the others. This is especially useful to see how the clusters created by PCA are reflected in the actual pricing/position of the announcements, the images below show how clearly announcements are clustered taking into consideration both the price and the neighbourhood:

images here

Moreover we also made it possible to select all the houses in a certain neighbourhood by selecting it in one of the barcharts; also this selection is obviously carried across all the graphs in the project.

8 Future work

While we are quite proud of the results that we were able to obtain with this project, one main thing that we have in mind and that for sure we will do but that is outside the scope of this exam is to create a “intelligent pricing” algorithm.

By this we mean a machine learning algorithm that is able, given the details of a new house, to estimate its optimal pricing, by taking into account similar announcements. We are quite confident that some powerful regressors like *Lasso* [10] or *Extreme Gradient Boosting* [11] would be able to extract the most important features and assign weights to them, so that a new house's price could be accurately estimated. This could of course be extremely useful for someone that is willing to add an announcement to the site but is unsure about the best price to put it at.

9 TODO

- Rimuovi TODO section lol
- Write Visualizations section
- Add images to Interaction design section
- Finalizza se hai usato tutto il dataset con node o se lo hai tagliato in sezione description of dataset
- prezzo per disponibiluita = income mensile
- selection across project
- fai ppt
- finalize layout

References

[1] “AirBnB’s website”
https://www.airbnb.it/.

[2] “Web traffic on AirBnB”
https://www.similarweb.com/website/airbnb.com.

[3] “Kaggle website’
https://www.kaggle.com/.

[4] “New York City AirBnB’s dataset’
https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data/.

[5] “Data-Driven Documents website’
https://d3js.org/.

[6] “PlotlyJS website’
https://plot.ly/javascript/.

[7] “Python website’
https://www.python.org/.

[8] “Pandas website’
https://pandas.pydata.org/.

[9] “jQuery website’
https://jquery.com/.

[10] “Lasso regression explained’
https://en.wikipedia.org/wiki/Lasso_(statistics).

[11] “Extreme Gradient Boosting documentation’
https://xgboost.readthedocs.io/en/latest/.