EUROPEAN CITIES SCORE

Calculate European cities score based on cost of living and number of venues around them

Alberto

May 20, 2020

Contents

1 Introduction

1.1 Background

Most of people like to travel. If you checkout online what are the best current trends, traveling is always in the top list. We have decided to make an application on this subject.

1.2 Problem

Travelers, when visiting new places, face several problems. Questions that may arise: what are the best place with lots of attractions, museums, coffee bars, restaurants can I visit and enjoy in my free time? Usually, we want to get the best deal we can. An important aspect to consider is the cost of living and other parameters for that particular city. We would prefer to get the highest quality at a reasonable price. This code will calculate, for every city in Europe, a score number. To calculate this score, we use cost of living, and number of venues nearby. The more venues and the lowest cost of living, the highest the score number. Cities with same score have the same color in a folium map. Green color means good scoring. Red color means bad scoring.

2 Data acquisition and cleaning

For the purpose, we have used this data source / services.

1) https://www.expatistan.com/cost-of-living/index/europe, where you can spot European cities cost of living
2) Foursquare that provides us information about venues near cities.

2.1 Data cleaning

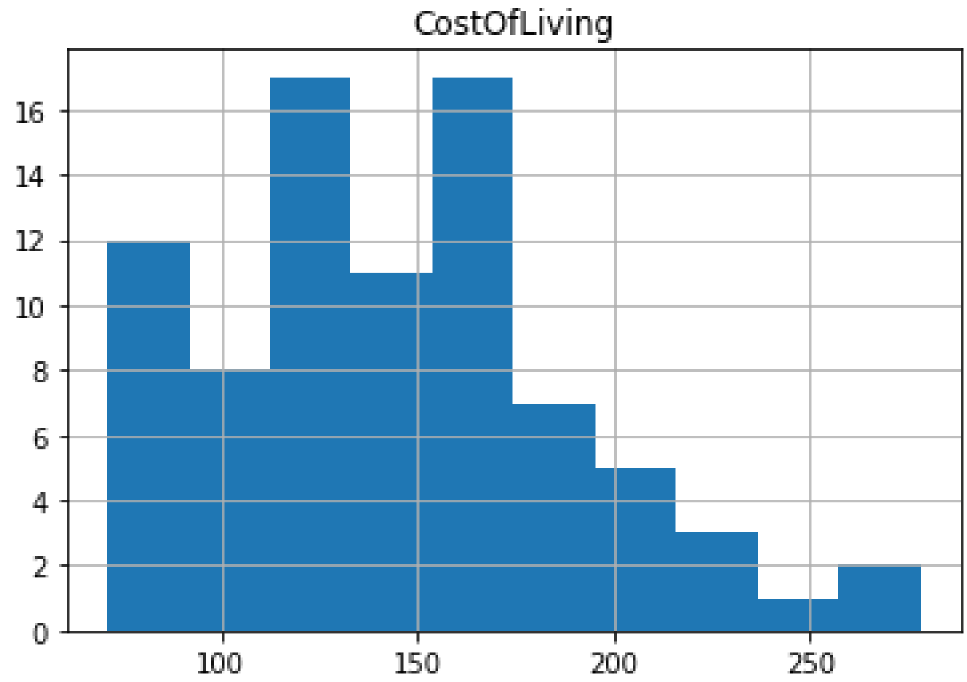
Scraping the website “cost of living”, data given was not always clean. Geolocator - used to spot map coordinates (latitude and longitude) of a particular city– not always returned correct data. So we have had to deal with “None” data in our pandas dataframe, and clean rows that contained corrupted data.

2.2 Feature selection

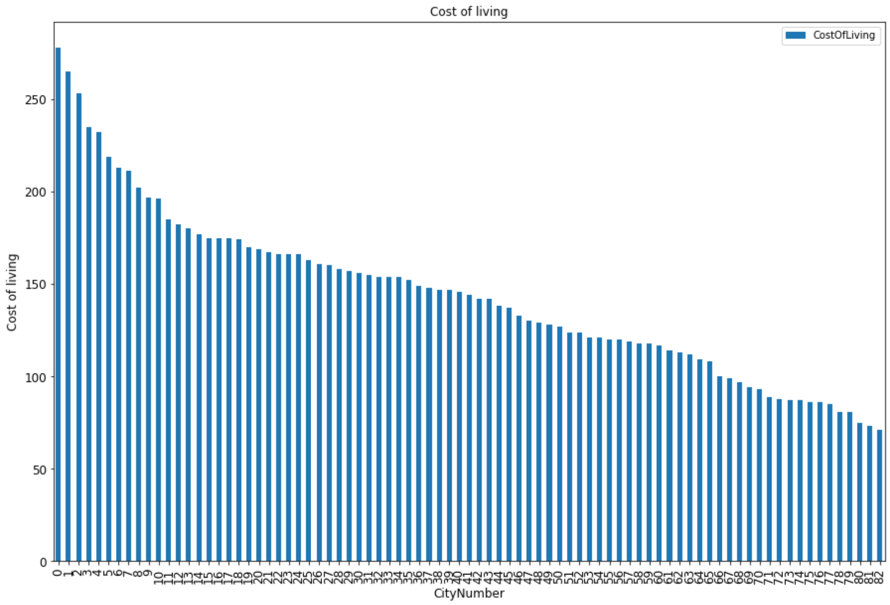
For every cities taken from the first source of data, we have used e geolocator to get coordinates of that city. We have used foursquare to spot venues using different keywords. In our purpose, the keyword was Italian. Foursquare spotted all venues containing the keyword “Italian”, using a distance of 1000 meters. Finally, we needed a dataframe that contained these features: CityName, Venues number, Invers cost of living, latitude, longitude, and final score. Invers cost of living is calculated as follows: (1 / cost of living). The lowest the cost of living, the better. Venues number is calculated by “groupy per CityName” and counting how many venues foursquare has find.

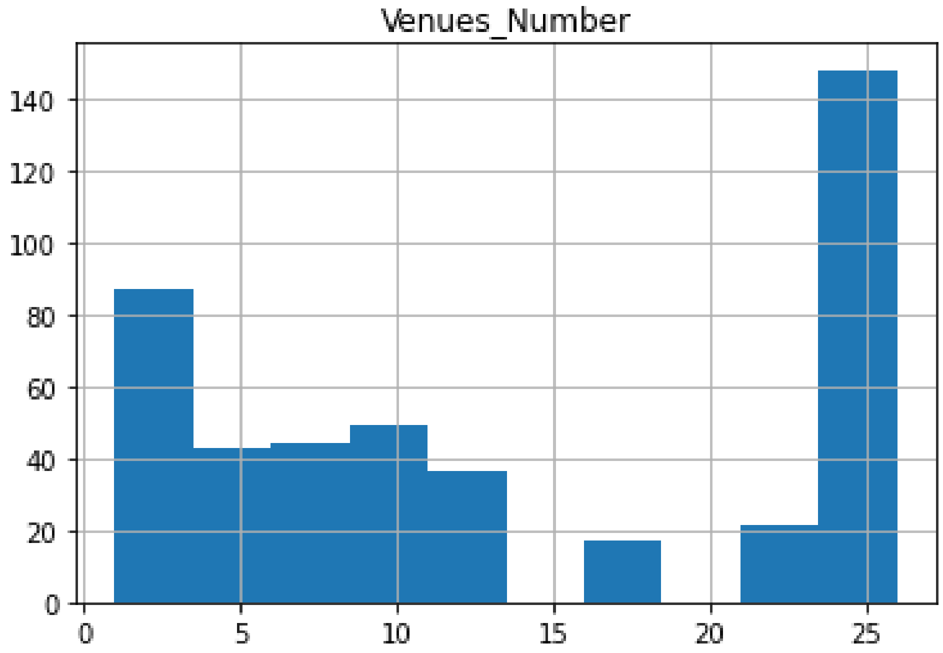
3 Data analysis

We can draw few charts that show data. For instance, below you can see the histogram of cost of living. Very few cities in Europe have very high cost of living (index above 200).



Always regarding cost of living, we can draw another char (bar chart) that displays cost of living of all European cities.





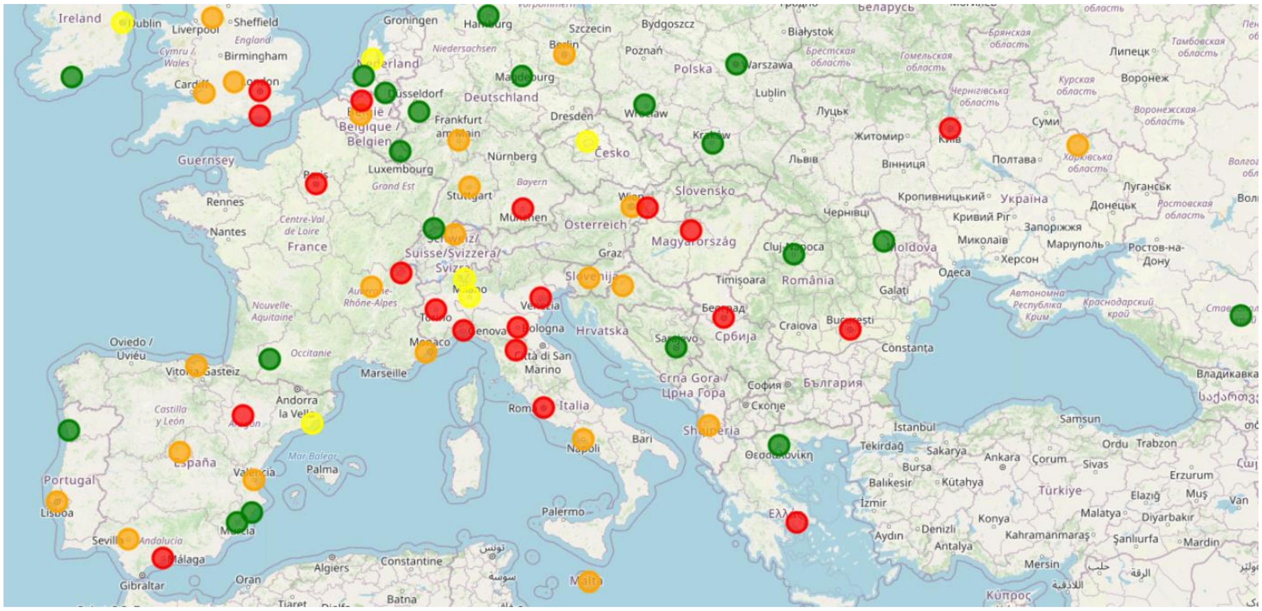
The chart above displays venues number per city. As you can see, many cities have many venues around them (1000meters)

4 Clustering model

For our program, we have used clustering approach (k means). In particular, we used library sklearn.cluster that provides already all functionalities we need. For k means, we have used two independent variables (InverseCostOfLiving and Venues_Number). Our "Scoring" dependent variable is calculated by k-means algorithm.

We have standardize features by removing the mean and scaling to unit variance.

Finally, to display the data in a map, we have used folium. The result can be seen below.



5 Conclusions

If we try to search venues using keyword "Italian", London is one of the best places in Europe. However, because it has a high cost of living, the score assigned to London is low. There are other places in Europe instead that have received a good scoring (eg; Germany, Spain), and that is because the low cost of living and a decent number of venues that are associated with Italian culture.