Real_estate_analysis

February 4, 2021

1 REAL ESTATE ANALYSIS: BUY/RENT HOUSES IN MILAN

1.1 How to select the best opportunities according to OMI price quotations and surrounding VENUES for buying houses in Milan.

1.1.1 Capstone Project - The Battle of Neighborhoods

Author: Pier Luigi Segatto, 02/02/2021 contact: pier.segatto@gmail.com

published on: https://pierluigi-segatto.medium.com/real-estate-analysis-252739af7c2b

2 NB the notebook itself is structured as a report

2.1 see table of contents.

2.2 Table of Contents

2.3 1. Introduction

Milan is the second-most populous city in Italy after Rome. The city proper has a population of about 1.4 million while its metropolitan city has 3.26 million inhabitants (ISTAT). Its continuously built-up urban area, that stretches well beyond the boundaries of the administrative metropolitan city, is the fourth largest in the EU with 5.27 million inhabitants. The population within the wider Milan metropolitan area, also known as Greater Milan, is estimated at 8.2 million, making it by far the largest metropolitan area in Italy and the 3rd largest in the EU (source). Milan is considered a leading global city, with strengths in the field of the art, commerce, design, education, entertainment, fashion, finance, healthcare, media, services, research and tourism. The city has been recognized as one of the world's four fashion capitals thanks to several international events and fairs, including Milan Fashion Week and the Milan Furniture Fair, which are currently among the world's biggest in terms of revenue, visitors and growth. It hosts numerous cultural institutions, academies and universities. Whereas Rome is Italy's political capital, Milan is the country's industrial and financial heart. In 2019 GDP per-capita of Milan is estimated at €49.000, steadily increasing, and significantly higher that the Italian average of €26.000 (source). Milan is the destination of 11 million visitors in 2019 (as reported in the city website (source), attracted by its museums and art galleries, that include some of the most important collections in the world, like the major works by Leonardo da Vinci. The city is served by many luxury hotels and dreamy restaurants. Last but not least, Milan will host the 2026 Winter Olympics together with Cortina d'Ampezzo.

2.4 1.1. Business Problem

Milan represents the epicenter for Italian life and it attracts companies, corporates, and people who move their core businesses and lives there. Due to the huge variety and heterogeneity of services and possibilities, prices for housing in Milan can be high and different among different areas of the city.

The goal of this project is to develop a tool for finding the most efficient *venue-* and *price*-wise solution for buying an house in Milan. This project will focus on finding the characteristics of each neighborhood in terms of house prices and relevant venues in the surrounding area (like restaurants, gyms, parks…). By adopting Machine learning solutions such as clustering and regression, this project will answer to the following questions:

1. If you want to buy or rent an house in Milan, which is the best neighborhood according to your capital, your lifestyle, and needs?

2. If you want to eat sushi and visit a museum, which neighborhood should you visit?

3. You are looking for an apartment, close to transportation station and to an italian restaurant, which neighborhood should you consider?

2.5 1.2. Target Audience

Real estates.

Housing investors.

Privates looking for the perfect place to rent or buy a house in Milan.

Tourists.

2.6 2. Data

The data for this project has been retrieved from multiple sources, paying the utmost attention to the reliability of them. For this reason, the data was collected from: 1. Section **??** and Section **??**: retrieved from the Italian Revenue Agency website (source), where the Milan borough list and the information about the market values and the rental values of the houses have been found, related to the 1st half of 2020, depending on the house location and the state of the property, and considering the negative influence brought by the COVID19 pandemic on real estate markets. In order to access to the CSV file, it's necessary to register to the website. 2. Section **??**: using Google Maps Geocoding API, it is possible to retrieve the geo-locational information (latitude and longitude) of Milan city center and the neighborhoods. 3. Section **??**: obtained using FourSquare API platform.

These datasets allow to explore and implement ML algorithms to gain insights on Milan and inform the final user on best locations. The Section **??** allowed to determine the value of the house, on the basis of the borough position and the state of the property. Neighborhoods locations have been fundamental to understand the correlation between the neighborhood positions (in terms of distance from the Milan city center) and the value of the houses. These positions, together with venues data, have been essential to determinate the clusters and identify the most common venues for each of them.

2.7 3. Methodology

In the following sections: - Libraries and external packages are loaded, Milan datasets are imported, cleaned and explored. - Neighborhoods' location are visualized and venues are downloaded and formatted to meet the required standards.

2.8 3.1. Requirements

The first and important step in data science is the data retrieval; indeed, there aren't reliable and precise analysis without using the best data and the most appropriate technique and algorithms. This analysis starts with the data collection and cleaning, in order to get all the essential data to achieve the goal of this study.

2.8.1 Download Libraries

uncomment the next cell if folium or geopy are not available

[1]:
```
# !conda install -c conda-forge folium=0.5.0 --yes
# !conda install -c conda-forge geopy --yes
```

Import the required libraries.

[2]:
```
import numpy as np # library to handle data in a vectorized manner

import pandas as pd # library for data analsysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

# Optimization and machine learning libraries
from scipy.optimize import curve_fit # fitting routines
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn import preprocessing # to normalize columns of dataframes

from geopy.geocoders import Nominatim # convert an address into latitude and␣
 ↪longitude values

import requests # library to handle requests

# Matplotlib and associated plotting modules
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import matplotlib.colors as colors
from matplotlib.colors import LinearSegmentedColormap

# Seaborn library for visualization
import seaborn as sns

import folium # map rendering library

print('Libraries imported.')
```

```
Libraries imported.
```

2.9 3.2. Milan boroughs dataset

Use the requests library to download the Milan dataset.

[3]:
```
url = "https://raw.githubusercontent.com/pierluigisegatto/Data_Science_Material/
 ↪main/IBM%20Data%20Science%20Professional%20Certificate/8-ML_Full_Project/
 ↪Real%20estate%20Milan%20/Milan%20Neighborhood.csv"
page = requests.get(url)
if page.status_code == 200:
    print('Page download successful')
else:
    print('Page download error. Error code: {}'.format(page.status_code))
```

```
Page download successful
```

[4]:
```
# be careful to use ';' as a separator
neighborhoods=pd.read_csv(url, sep=';')
neighborhoods.head(5)
```

[4]:
```
  Area_territoriale    Regione Prov  Comune_ISTAT Comune_cat  Sez Comune_amm  \
0        NORD-OVEST  LOMBARDIA   MI       3015146       C1AA  NaN       F205
1        NORD-OVEST  LOMBARDIA   MI       3015146       C1AA  NaN       F205
2        NORD-OVEST  LOMBARDIA   MI       3015146       C1AA  NaN       F205
3        NORD-OVEST  LOMBARDIA   MI       3015146       C1AA  NaN       F205
4        NORD-OVEST  LOMBARDIA   MI       3015146       C1AA  NaN       F205

  Comune_descrizione Fascia  \
0             MILANO      B
1             MILANO      B
2             MILANO      B
3             MILANO      B
4             MILANO      B

                                          Zona_Descr Zona    LinkZona  \
0  'CENTRO STORICO -DUOMO, SANBABILA, MONTENAPOLE…  B12  MI00003228
1              'CENTRO STORICO -UNIVERSITA STATALE'  B13  MI00003232
2                         'CENTRO STORICO - BRERA'  B15  MI00003544
3  'CENTRO STORICO -SANT`AMBROGIO, CADORNA, VIA D…  B16  MI00003545
4                 'PARCO SEMPIONE, ARCO DELLA PACE'  B17  MI00004767

   Cod_tip_prev     Descr_tip_prev Stato_prev  Microzona
0            20  Abitazioni civili          N          2
1            20  Abitazioni civili          N          3
2            20  Abitazioni civili          0          0
3            20  Abitazioni civili          0          0
4            20  Abitazioni civili          N          0
```

[5]:
```
# understand a bit the dataset
print(f'Minal has {neighborhoods.shape[0]} Boroughs')
```

```
Minal has 38 Boroughs
```

[6]:
```
# check for unique values
print(neighborhoods['Descr_tip_prev'].unique())
print(neighborhoods['Stato_prev'].unique())
print(neighborhoods['Microzona'].unique())
```

```
['Abitazioni civili' 'Non presente']
['N' '0' nan]
[ 2  3  0 22 32 34 35 37 40 42 43 47 50]
```

[7]:
```
# where the anomaly occur?
neighborhoods.loc[(neighborhoods['Descr_tip_prev']== 'Non presente') |␣
 ↪(neighborhoods['Stato_prev']== np.nan),'Zona_Descr']
```

[7]:
```
37    'RONCHETTO, CHIARAVALLE, RIPAMONTI'
Name: Zona_Descr, dtype: object
```

Filter only needed columns, i.e. keep only the neighborhood description name the zone code. Translate column names and clean the neighborhoods names from quotes.

Get the actual Neighborhood names by splitting each descriptive name (Boro) where comma occurs. This step is needed to retrieve the latitude and longitude of each neighborhoods.

[8]:
```
# keep useful cols
milan = neighborhoods[['Zona_Descr','Zona','Fascia']].copy()
# translate col names
milan.rename(columns={'Zona_Descr':'Neighborhoods',"Zona":'Code','Fascia':
 ↪'Area'},inplace=True)
# Get rid of quotes in names
milan['Neighborhoods'].replace("'", '',regex=True,inplace=True)
# explode the dataset to get the actual neighborhood names
milan = milan.set_index(['Code','Area']).apply(lambda x: x.str.split(',').
 ↪explode()).reset_index()
# porta romana occurs 2 times one with code B19 and one with code B20
milan[['Neighborhoods']] = milan[['Neighborhoods']].
 ↪drop_duplicates(keep='first')

milan.head(5)
```

[8]:

	Code	Area	Neighborhoods
0	B12	B	CENTRO STORICO -DUOMO
1	B12	B	SANBABILA
2	B12	B	MONTENAPOLEONE
3	B12	B	MISSORI
4	B12	B	CAIROLI

[9]:
```
print(f'Milan has {milan.shape[0]} Neighborhoods')
```

```
Milan has 104 Neighborhoods
```

Some neighborhood names cannot be recognized by Google Maps Geocoding API. It's essential to edit them, to ensure the success of the last step (latitudes and longitudes retrive)

[10]:
```
milan['Neighborhoods'].replace("SANBABILA", "SAN BABILA", regex=True,␣
 ↪inplace=True)
milan['Neighborhoods'].replace("FAENZA", "VIALE FAENZA", regex=True,␣
 ↪inplace=True)
```

```
milan['Neighborhoods'].replace("P.ROSSI", "VIA PELLEGRINO ROSSI", regex=True,␣
↪inplace=True)
milan['Neighborhoods'].replace("CORSO VENEZIA", "PORTA VENEZIA", regex=True,␣
↪inplace=True)
milan['Neighborhoods'].replace("SANTA GIULIA", "MILANO SANTA GIULIA",␣
↪regex=True, inplace=True)
milan['Neighborhoods'].replace("UNIVERSITA STATALE", "CITTÀ STUDI", regex=True,␣
↪inplace=True)
milan['Neighborhoods'].replace("STAZIONE CENTRALE VIALE STELVIO", "STAZIONE␣
↪CENTRALE", regex=True, inplace=True)
milan['Neighborhoods'].replace("C.NA MERLATA", "CASCINA MERLATA", regex=True,␣
↪inplace=True)
milan['Neighborhoods'].replace("MONZA", "VIALE MONZA", regex=True, inplace=True)
milan['Neighborhoods'].replace("BUENOS AIRES", "CORSO BUENOS AIRES",␣
↪regex=True, inplace=True)
milan['Neighborhoods'].replace("TITO LIVIO", "VIA TITO LIVIO", regex=True,␣
↪inplace=True)
milan['Neighborhoods'].replace("MAROCCHETTI", "VIA CARLO MAROCHETTI",␣
↪regex=True, inplace=True)
milan['Neighborhoods'].replace("REGINA GIOVANNA", "VIALE REGINA GIOVANNA",␣
↪regex=True, inplace=True)
milan['Neighborhoods'].replace("ASCANIO SFORZA", "VIA ASCANIO SFORZA",␣
↪regex=True, inplace=True)
milan['Neighborhoods'].replace("Q. ROMANO", "QUINTO ROMANO", regex=True,␣
↪inplace=True)

milan['Neighborhoods'] = milan['Neighborhoods'].str.replace('CENTRO STORICO -',␣
↪'')
```

[11]:
```
# Remove Venezia as there is a duplicate called PORTA VENEZIA and VENEZIA
milan = milan[milan['Neighborhoods'] != "VENEZIA"]
milan.shape
```

[11]: (103, 3)

[12]:
```
# remove all white spaces at beginning of each neigh name
milan['Neighborhoods'] = milan['Neighborhoods'].str.lstrip(' ')
```

[13]:
```
neigh_df = milan[['Neighborhoods']]
print(f'Milan has {neigh_df.shape[0]} unique Neighborhood names')
```

```
Milan has 103 unique Neighborhood names
```

2.9.1 Retrive the geo-locational information

[14]:
```
from geopy.exc import GeocoderTimedOut
from geopy.exc import GeocoderNotFound

address= (neigh_df['Neighborhoods'] + ', Milano, MI , Italia')
geolocater= Nominatim(user_agent="milan_coordinates-explorer")
location=[]
empty=[]

def getcoords(add):
    try:
        coords= geolocater.geocode(add, timeout=10)
        location.append([add, coords.latitude, coords.longitude])
        print("the coords are {}".format(location[-1]))

    except GeocoderTimedOut:
        return getcoords(add)

    except:
        empty.append([add])
        print("Couldn't find coords of {}".format(empty[-1]))

for add in address:
        getcoords(add)
```

```
the coords are ['DUOMO, Milano, MI , Italia', 45.4645848, 9.1896695]
the coords are ['SAN BABILA, Milano, MI , Italia', 45.4665214, 9.1975286]
the coords are ['MONTENAPOLEONE, Milano, MI , Italia', 45.470015, 9.1928678]
the coords are ['MISSORI, Milano, MI , Italia', 45.4598278, 9.1895549]
the coords are ['CAIROLI, Milano, MI , Italia', 45.4687012, 9.1816966]
the coords are ['CITTÀ STUDI, Milano, MI , Italia', 45.4770557, 9.2265746]
the coords are ['BRERA, Milano, MI , Italia', 45.47347885, 9.188407990372653]
the coords are ['SANT`AMBROGIO, Milano, MI , Italia', 45.4613906, 9.1729167]
the coords are ['CADORNA, Milano, MI , Italia', 45.4681551, 9.1771024]
the coords are ['VIA DANTE, Milano, MI , Italia', 45.4663326, 9.1847796]
the coords are ['PARCO SEMPIONE, Milano, MI , Italia', 45.47301905,
9.176969268773153]
the coords are ['ARCO DELLA PACE, Milano, MI , Italia', 45.47569195,
9.172427802834267]
the coords are ['TURATI, Milano, MI , Italia', 45.475039, 9.1947243]
the coords are ['MOSCOVA, Milano, MI , Italia', 45.4770906, 9.1843416]
the coords are ['PORTA VENEZIA, Milano, MI , Italia', 45.474282099999996,
9.205226499999997]
the coords are ['PORTA VITTORIA, Milano, MI , Italia', 45.4622607, 9.2095796]
the coords are ['PORTA ROMANA, Milano, MI , Italia', 45.4522392, 9.20205595]
the coords are ['PORTA VIGENTINA, Milano, MI , Italia', 45.451351, 9.1964451]
```

```
the coords are [nan, 46.3144754, 11.0480288]
the coords are ['PORTA TICINESE, Milano, MI , Italia', 45.45248575,
9.18020677095896]
the coords are ['PORTA GENOVA, Milano, MI , Italia', 45.4566646, 9.174606]
the coords are ['CORSO BUENOS AIRES, Milano, MI , Italia', 45.4775766,
9.2082162]
the coords are ['VIALE REGINA GIOVANNA, Milano, MI , Italia', 45.4738157,
9.2132568]
the coords are ['CITY LIFE, Milano, MI , Italia', 45.4833239, 9.2022214]
the coords are ['PORTA NUOVA, Milano, MI , Italia', 45.4798577,
9.192694240094355]
the coords are ['STAZIONE CENTRALE, Milano, MI , Italia', 45.4865088, 9.2073553]
the coords are ['CENISIO, Milano, MI , Italia', 45.4875646, 9.1722651]
the coords are ['FARINI, Milano, MI , Italia', 45.49287940000001,
9.175108231228549]
the coords are ['SARPI, Milano, MI , Italia', 45.4830418, 9.172131735619073]
the coords are ['SEMPIONE, Milano, MI , Italia', 45.4771041, 9.1701771]
the coords are ['PAGANO, Milano, MI , Italia', 45.4682848, 9.1610998]
the coords are ['WASHINGTON, Milano, MI , Italia', 45.4612061, 9.15631041155902]
the coords are ['SOLARI, Milano, MI , Italia', 45.4590225, 9.1648878]
the coords are ['P.TA GENOVA, Milano, MI , Italia', 45.4566646, 9.174606]
the coords are ['VIA ASCANIO SFORZA, Milano, MI , Italia', 45.4500128,
9.1777002]
the coords are ['TABACCHI, Milano, MI , Italia', 45.4867765, 9.1982143]
the coords are ['SARFATTI, Milano, MI , Italia', 45.4482692, 9.1891795]
the coords are ['PARCO LAMBRO, Milano, MI , Italia', 45.496842799999996,
9.250524307072563]
the coords are ['FELTRE, Milano, MI , Italia', 45.4892131, 9.2461182]
the coords are ['UDINE, Milano, MI , Italia', 45.4912044, 9.2369964]
the coords are ['PIOLA, Milano, MI , Italia', 45.481145, 9.2259168]
the coords are ['ARGONNE, Milano, MI , Italia', 45.4670479, 9.2303962]
the coords are ['CORSICA, Milano, MI , Italia', 45.463908599999996,
9.23080179796616]
the coords are ['LAMBRATE, Milano, MI , Italia', 45.4831483, 9.2419983]
the coords are ['RUBATTINO, Milano, MI , Italia', 45.4774421, 9.2523668]
the coords are ['ROMBON, Milano, MI , Italia', 45.4864231, 9.2409806]
the coords are ['FORLANINI, Milano, MI , Italia', 45.4675255, 9.2135715]
the coords are ['MECENATE, Milano, MI , Italia', 45.4477791, 9.247090719928508]
the coords are ['ORTOMERCATO, Milano, MI , Italia', 45.45330395,
9.23059287245498]
the coords are ['MILANO SANTA GIULIA, Milano, MI , Italia', 45.4368295,
9.2437113]
the coords are ['VIA CARLO MAROCHETTI, Milano, MI , Italia', 45.4393538,
9.2252418]
the coords are ['VIGENTINO, Milano, MI , Italia', 45.4351394, 9.1986952]
the coords are ['CHIESA ROSSA, Milano, MI , Italia', 45.4277094, 9.17457]
the coords are ['ORTLES, Milano, MI , Italia', 45.4391034, 9.2014801]
the coords are ['SPADOLINI, Milano, MI , Italia', 45.4422653, 9.1920689]
```

```
the coords are ['BAZZI, Milano, MI , Italia', 45.442518, 9.1880159]
the coords are ['BARONA, Milano, MI , Italia', 45.4307236, 9.153586498645359]
the coords are ['FAMAGOSTA, Milano, MI , Italia', 45.436895, 9.1680126]
the coords are ['VIALE FAENZA, Milano, MI , Italia', 45.4369963, 9.1505196]
the coords are ['LORENTEGGIO, Milano, MI , Italia', 45.4458965, 9.1263786]
the coords are ['INGANNI, Milano, MI , Italia', 45.457563, 9.1222629]
the coords are ['BISCEGLIE, Milano, MI , Italia', 45.4554126, 9.1129589]
the coords are ['SAN CARLO B., Milano, MI , Italia', 45.4921361, 9.2289488]
the coords are ['IPPODROMO, Milano, MI , Italia', 45.4761386, 9.1290346]
the coords are ['CAPRILLI, Milano, MI , Italia', 45.4801917, 9.1329165]
the coords are ['MONTE STELLA, Milano, MI , Italia', 45.4905858, 9.1346218]
the coords are ['MUSOCCO, Milano, MI , Italia', 45.5093132, 9.1319408]
the coords are ['CERTOSA, Milano, MI , Italia', 45.5024894, 9.1278477]
the coords are ['EXPO, Milano, MI , Italia', 45.5229171, 9.0940419]
the coords are ['CASCINA MERLATA, Milano, MI , Italia', 45.508611, 9.104424]
the coords are ['BOVISA, Milano, MI , Italia', 45.5027696, 9.1612638]
the coords are ['BAUSAN, Milano, MI , Italia', 45.5026343, 9.1663739]
the coords are ['IMBONATI, Milano, MI , Italia', 45.4388011, 9.20482096513496]
the coords are ['BOVISASCA, Milano, MI , Italia', 45.5158419, 9.1537779]
the coords are ['AFFORI, Milano, MI , Italia', 45.5170295, 9.1696533]
the coords are ['P. ROSSI , Milano, MI , Italia', 44.91709965,
9.789736517763782]
the coords are ['COMASINA, Milano, MI , Italia', 45.5281326, 9.1637291]
the coords are ['NIGUARDA, Milano, MI , Italia', 45.5169738, 9.1924879]
the coords are ['BIGNAMI, Milano, MI , Italia', 45.5267842, 9.2121805]
the coords are ['PARCO NORD, Milano, MI , Italia', 45.51994475,
9.181313083892288]
the coords are ['SARCA, Milano, MI , Italia', 45.50521689999999,
9.203916460372216]
the coords are ['BICOCCA, Milano, MI , Italia', 45.5149166, 9.2111381]
the coords are ['VIALE MONZA, Milano, MI , Italia', 45.51248185,
9.22433270585019]
the coords are ['CRESCENZAGO, Milano, MI , Italia', 45.5092191, 9.2474843]
the coords are ['GORLA, Milano, MI , Italia', 45.5049445, 9.2245393]
the coords are ['QUARTIERE ADRIANO, Milano, MI , Italia', 45.5160356, 9.2444433]
the coords are ['MAGGIOLINA, Milano, MI , Italia', 45.4920365, 9.2018846]
the coords are ['PARCO TROTTER, Milano, MI , Italia', 45.494495900000004,
9.22419935442353]
the coords are ['LEONCAVALLO, Milano, MI , Italia', 45.490367, 9.2247833]
the coords are ['BAGGIO, Milano, MI , Italia', 45.4613839, 9.089843]
the coords are ['QUINTO ROMANO, Milano, MI , Italia', 45.4756791, 9.0888066]
the coords are ['MUGGIANO, Milano, MI , Italia', 45.4510311, 9.0697581]
the coords are ['GALLARATESE, Milano, MI , Italia', 45.4980249, 9.1086684]
the coords are ['LAMPUGNANO, Milano, MI , Italia', 45.4866108, 9.1250144]
the coords are ['P. TRENNO, Milano, MI , Italia', 45.4906416, 9.1008711]
the coords are ['BONOLA, Milano, MI , Italia', 45.4971385, 9.1098414]
the coords are ['MISSAGLIA, Milano, MI , Italia', 45.7087169, 9.335754]
the coords are ['GRATOSOGLIO, Milano, MI , Italia', 45.407321, 9.1712557]
```

```
the coords are ['QUARTO OGGIARO, Milano, MI , Italia', 45.5143752, 9.1406811]
the coords are ['SACCO, Milano, MI , Italia', 45.5203654, 9.123898569900575]
the coords are ['RONCHETTO, Milano, MI , Italia', 45.525659950000005,
9.382790861222784]
the coords are ['CHIARAVALLE, Milano, MI , Italia', 45.4166968, 9.2374208]
the coords are ['RIPAMONTI, Milano, MI , Italia', 45.42851865,
9.203143715025023]
```

2.9.2 Save found neighborhood coordinates

[15]:
```
neighborhoods_coordinates = pd.DataFrame(location,␣
↪columns=['Neighborhoods','Latitude','Longitude'])
neighborhoods_coordinates['Neighborhoods'].replace(", Milano, MI , Italia", "",␣
↪regex=True, inplace=True)
neighborhoods_coordinates['Area'] = milan['Area']
neighborhoods_coordinates.to_csv('coordinates.csv')

neighborhoods_coordinates.head()
```

[15]:

	Neighborhoods	Latitude	Longitude	Area
0	DUOMO	45.464585	9.189670	B
1	SAN BABILA	45.466521	9.197529	B
2	MONTENAPOLEONE	45.470015	9.192868	B
3	MISSORI	45.459828	9.189555	B
4	CAIROLI	45.468701	9.181697	B

2.10 3.3 Milan neighborhoods visualization

To get a sense to the study, it is of primary importance to know the precise location of each neighborhoods. For this reason, it's essential to create a map of Milan, in which all the neighborhood positions are shown. To be more exhaustive, all the Milan areas (B, C, D and E) are differentiated by different colors. Therefore, the map of Milan neighborhoods has been plotted using the Folium library.

2.10.1 Retrieve Milan coordinates

[16]:
```
Milan_address='Milan, Italy'
geolocater= Nominatim(user_agent="Milan_search")
center= geolocater.geocode(Milan_address)
lat= center.latitude
lon= center.longitude
print('The geograpical coordinate of {} are {}, {}.'.format(Milan_address,lat,␣
↪lon))
```

```
The geograpical coordinate of Milan, Italy are 45.4668, 9.1905.
```

2.10.2 Create the map

[17]:
```
neighborhoods_coordinates['Area'] = milan['Code'].str[:1]
neighborhoods_coordinates['Code'] = milan['Code']
neighborhoods_coordinates.head()
```

[17]:

	Neighborhoods	Latitude	Longitude	Area	Code
0	DUOMO	45.464585	9.189670	B	B12
1	SAN BABILA	45.466521	9.197529	B	B12
2	MONTENAPOLEONE	45.470015	9.192868	B	B12
3	MISSORI	45.459828	9.189555	B	B12
4	CAIROLI	45.468701	9.181697	B	B12

[18]:
```
Milan_map=folium.Map(location=[lat,lon],zoom_start=11)

def color(letter):
    if letter == 'B':
        col = 'blue'
    elif letter == 'C':
        col = 'green'
    elif letter == 'D':
        col = 'purple'
    else:
        col='red'
    return col

for lat, long, nieghborhood, letter in␣
 ↪zip(neighborhoods_coordinates['Latitude'],␣
 ↪neighborhoods_coordinates['Longitude'],␣
 ↪neighborhoods_coordinates['Neighborhoods'],␣
 ↪neighborhoods_coordinates['Area']):
    label=folium.Popup(nieghborhood, parse_html=True)
    folium.CircleMarker(
        [lat, long],
        radius=2,
        popup=label,
        color= color(letter),
        fill=True,
        fill_color='#3199cc',
        fill_opacity=0.3,
        parse_html=False, legend_name='SCALE').add_to(Milan_map)

legend_html = '''
    <div style="position: fixed;
        bottom: 10px; left: 10px; width: 100px; height: 120px;
        border:2px solid grey; z-index:9999; font-size:11px;">  Legend <br>
```

```
          <b> Area B </b>   <i class="fa fa-map-marker fa-2x"␣
↪style="color:blue"></i><br>
          <b> Area C </b>   <i class="fa fa-map-marker fa-2x"␣
↪style="color:green"></i><br>
          <b> Area D </b>   <i class="fa fa-map-marker fa-2x"␣
↪style="color:purple"></i><br>
          <b> Area E </b>   <i class="fa fa-map-marker fa-2x"␣
↪style="color:red"></i><br>
    </div>
    '''
Milan_map.get_root().html.add_child(folium.Element(legend_html))

Milan_map
```

[18]: <folium.folium.Map at 0x1a18d50a58>

2.11 3.4 OMI quotations housing price dataset

[19]:
```
url = "https://raw.githubusercontent.com/pierluigisegatto/Data_Science_Material/
↪main/IBM%20Data%20Science%20Professional%20Certificate/8-ML_Full_Project/
↪Real%20estate%20Milan%20/Market%20values%20of%20house%20in%20Milan.csv"
page = requests.get(url)
if page.status_code == 200:
    print('Page download successful')
else:
    print('Page download error. Error code: {}'.format(page.status_code))
```

```
Page download successful
```

[20]:
```
values=pd.read_csv(url, sep=';')
values.head(10)
```

[20]:

	Area_territoriale	Regione	Prov	Comune_ISTAT	Comune_cat	Sez	Comune_amm	\
0	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
1	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
2	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
3	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
4	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
5	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
6	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
7	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
8	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	
9	NORD-OVEST	LOMBARDIA	MI	3015146	C1AA		F205	

	Comune_descrizione	Fascia	Zona	LinkZona	Cod_Tip	\

```
0             MILANO      B  B12  MI00003228       20
1             MILANO      B  B12  MI00003228       20
2             MILANO      B  B12  MI00003228       21
3             MILANO      B  B12  MI00003228       21
4             MILANO      B  B12  MI00003228       19
5             MILANO      B  B12  MI00003228       13
6             MILANO      B  B12  MI00003228        9
7             MILANO      B  B12  MI00003228        5
8             MILANO      B  B12  MI00003228        5
9             MILANO      B  B12  MI00003228        6

                Descr_Tipologia    Stato Stato_prev  Compr_min  Compr_max  \
0              Abitazioni civili   OTTIMO        NaN       9000      12300
1              Abitazioni civili  NORMALE          P       7400       9000
2  Abitazioni di tipo economico  NORMALE          P       6500       7800
3  Abitazioni di tipo economico   OTTIMO        NaN       7800       9000
4           Abitazioni signorili   OTTIMO          P      11200      14300
5                            Box  NORMALE          P       4700       6600
6                     Magazzini  NORMALE          P       2550       3300
7                         Negozi   OTTIMO        NaN      15100      21600
8                         Negozi  NORMALE          P       8800      12400
9                         Uffici   OTTIMO          P       5800       9000

  Sup_NL_compr Loc_min Loc_max Sup_NL_loc
0            L      28    37,5          L
1            L      23      28          L
2            L      18      22          L
3            L      24      30          L
4            L    37,5      46          L
5            L    14,5    21,5          L
6            L      14      18          L
7            L      83     124          L
8            L      40      59          L
9            L    21,3      36          L
```

This dataset presents for each zone and for each housing type and status (if existing) combination the average observed price.

[21]:
```
values.shape
```

[21]:
```
(484, 21)
```

[22]:
```
# check the housing types available
values['Descr_Tipologia'].unique()
```

[22]:
```
array(['Abitazioni civili', 'Abitazioni di tipo economico',
       'Abitazioni signorili', 'Box', 'Magazzini', 'Negozi', 'Uffici',
```

```
       'Uffici strutturati', 'Laboratori', 'Capannoni industriali',
       'Capannoni tipici', 'Ville e Villini'], dtype=object)
```

In order to merge this dataset with the neighborhoods I am interested in collecting the zone code along with the housing description and prices.

[23]:
```
columns =␣
 ↪['Zona','Descr_Tipologia','Stato','Compr_min','Compr_max','Loc_min','Loc_max']
omi = values[columns].copy()
omi.head()
```

[23]:

	Zona	Descr_Tipologia	Stato	Compr_min	Compr_max	Loc_min	Loc_max
0	B12	Abitazioni civili	OTTIMO	9000	12300	28	37,5
1	B12	Abitazioni civili	NORMALE	7400	9000	23	28
2	B12	Abitazioni di tipo economico	NORMALE	6500	7800	18	22
3	B12	Abitazioni di tipo economico	OTTIMO	7800	9000	24	30
4	B12	Abitazioni signorili	OTTIMO	11200	14300	37,5	46

N.B: Since I am interested in analyzing houses, I drop all occurrences of non living places.

[24]:
```
# translate column headers
omi.rename(columns={'Zona' : 'Code', 'Descr_Tipologia' : 'Housing_type',␣
 ↪'Stato' : 'Condition', 'Compr_min' : 'Min_market_value (€/m2)', 'Compr_max' :␣
 ↪'Max_market_value (€/m2)', 'Loc_min' : 'Min_rental_value (€/m2 x month)',␣
 ↪'Loc_max' : 'Max_rental_value (€/m2 x month)',}, inplace=True)
print(omi.shape)

# drop non-living quotations
omi = omi[(omi['Housing_type'] == 'Abitazioni civili') | (omi['Housing_type']␣
 ↪== 'Abitazioni signorili')]
# Translate to english housing types, condition values
omi = omi.replace(regex={'OTTIMO': 'Excellent','NORMALE':'Normal','Abitazioni␣
 ↪civili':'Residential houses','Abitazioni signorili':'Stately houses'})
print(omi.shape)
omi.head()
```

```
(484, 7)
(88, 7)
```

[24]:
```
   Code        Housing_type  Condition  Min_market_value (€/m2)  \
0   B12  Residential houses  Excellent                     9000
1   B12  Residential houses     Normal                     7400
4   B12      Stately houses  Excellent                    11200
12  B13  Residential houses  Excellent                     6900
13  B13  Residential houses     Normal                     5000

    Max_market_value (€/m2) Min_rental_value (€/m2 x month)  \
0                     12300                              28
1                      9000                              23
4                     14300                            37,5
12                     8200                            18,5
13                     6900                            14,5

   Max_rental_value (€/m2 x month)
0                             37,5
1                               28
4                               46
12                            27,3
13                            18,5
```

2.11.1 Adjust the data type of the dataframe

[25]:
```
omi.dtypes
```

[25]:
```
Code                                object
Housing_type                        object
Condition                           object
Min_market_value (€/m2)              int64
Max_market_value (€/m2)              int64
Min_rental_value (€/m2 x month)     object
Max_rental_value (€/m2 x month)     object
dtype: object
```

We can note that the data type of Min_rental_value and Max_rental_value columns are not correct. For this reason we will transform the columns data type to float. The problem is that decimals are preceded by comma and not by dot. So first we will replace ',' with '.'

[26]:
```
columns_to_change = ['Min_rental_value (€/m2 x month)','Max_rental_value (€/m2␣
↪x month)']
omi[columns_to_change]=omi[columns_to_change].replace(regex={',':'.'})

omi[columns_to_change] = omi[columns_to_change].astype('float')

omi.head()
```

[26]:
```
   Code        Housing_type  Condition  Min_market_value (€/m2)  \
0   B12  Residential houses  Excellent                     9000
1   B12  Residential houses     Normal                     7400
4   B12      Stately houses  Excellent                    11200
12  B13  Residential houses  Excellent                     6900
13  B13  Residential houses     Normal                     5000

    Max_market_value (€/m2)  Min_rental_value (€/m2 x month)  \
0                     12300                             28.0
1                      9000                             23.0
4                     14300                             37.5
12                     8200                             18.5
13                     6900                             14.5

    Max_rental_value (€/m2 x month)
0                              37.5
1                              28.0
4                              46.0
12                             27.3
13                             18.5
```

[27]:
```
omi.dtypes
```

[27]:
```
Code                                object
Housing_type                        object
Condition                           object
Min_market_value (€/m2)              int64
Max_market_value (€/m2)              int64
Min_rental_value (€/m2 x month)    float64
Max_rental_value (€/m2 x month)    float64
dtype: object
```

2.11.2 Merge the neighborhoods_coordinates and omi dataframes on borough codes

[28]:
```
full_df = pd.merge(neighborhoods_coordinates, omi, on='Code')
full_df.head()
```

[28]:
```
  Neighborhoods   Latitude  Longitude Area Code        Housing_type  \
0         DUOMO  45.464585   9.189670    B  B12  Residential houses
1         DUOMO  45.464585   9.189670    B  B12  Residential houses
2         DUOMO  45.464585   9.189670    B  B12      Stately houses
3    SAN BABILA  45.466521   9.197529    B  B12  Residential houses
4    SAN BABILA  45.466521   9.197529    B  B12  Residential houses

   Condition  Min_market_value (€/m2)  Max_market_value (€/m2)  \
0  Excellent                     9000                    12300
```

```
1     Normal                    7400                    9000
2  Excellent                   11200                   14300
3  Excellent                    9000                   12300
4     Normal                    7400                    9000

   Min_rental_value (€/m2 x month)  Max_rental_value (€/m2 x month)
0                             28.0                             37.5
1                             23.0                             28.0
2                             37.5                             46.0
3                             28.0                             37.5
4                             23.0                             28.0
```

[29]:
```
full_df.shape
```

[29]:
```
(218, 11)
```

2.12 3.5 Neighborhood venues

2.12.1 Define Foursquare credentials and version

[30]:
```
CLIENT_ID = '5HAYNQMXEEGNMBKVOEKKQUEVKGHGSLA5H2A4YULOKF3S3TGC' # your␣
 ↪Foursquare ID
CLIENT_SECRET = 'JHVYWCNPODUHLPTLKQL3FW5ZU11KVSGDJOTYZURAWWA5QMD4' # your␣
 ↪Foursquare Secret
VERSION = '20180605' # Foursquare API version
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: 5HAYNQMXEEGNMBKVOEKKQUEVKGHGSLA5H2A4YULOKF3S3TGC
CLIENT_SECRET:JHVYWCNPODUHLPTLKQL3FW5ZU11KVSGDJOTYZURAWWA5QMD4
```

2.12.2 Get the top 100 venues for each neighborhoods within a radius of 1 Km.

[31]:
```
radius = 1000
LIMIT = 100

venues = []

for lat, long, neighborhood in zip(neighborhoods_coordinates['Latitude'],␣
 ↪neighborhoods_coordinates['Longitude'],␣
 ↪neighborhoods_coordinates['Neighborhoods']):
    url = "https://api.foursquare.com/v2/venues/explore?
 ↪client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}".format(
```

```
        CLIENT_ID,
        CLIENT_SECRET,
        VERSION,
        lat,
        long,
        radius,
        LIMIT)

    results = requests.get(url).json()["response"]['groups'][0]['items']

    for venue in results:
        venues.append((
            neighborhood,
            lat,
            long,
            venue['venue']['name'],
            venue['venue']['location']['lat'],
            venue['venue']['location']['lng'],
            venue['venue']['categories'][0]['name']))
```

[32]:
```
# convert the venues list into a new DataFrame
milan_venues = pd.DataFrame(venues)

milan_venues.columns = ['Neighborhoods', 'Latitude', 'Longitude', 'Venue_Name',␣
↪'Venue_Latitude', 'Venue_Longitude', 'Venue_Category']

print(milan_venues.shape)
milan_venues.head()
```

```
(6745, 7)
```

[32]:

	Neighborhoods	Latitude	Longitude	Venue_Name	Venue_Latitude	Venue_Longitude	Venue_Category
0	DUOMO	45.464585	9.18967	Piazza del Duomo	45.464190	9.189527	Plaza
1	DUOMO	45.464585	9.18967	Galleria Vittorio Emanuele II	45.465577	9.190024	Monument / Landmark
2	DUOMO	45.464585	9.18967	Room Mate Giulia Hotel	45.465250	9.189396	Hotel
3	DUOMO	45.464585	9.18967	Terrazze del Duomo	45.464207	9.191075	Scenic Lookout
4	DUOMO	45.464585	9.18967	Palazzo Reale	45.462960	9.191348	Art Gallery

2.13 3.6. Data visualization: quotation prices for each neighborhood

Bar charts have been created to determine the house price behaviors of each Milan borough, depending also on the housing type (residential or stately house) and the condition of residential homes (excellent or normal).

2.13.1 Create a column for the average market and rental values

[33]:
```
full_df['Avg_market_value (€/m2)'] = full_df[['Min_market_value (€/m2)',␣
↪'Max_market_value (€/m2)']].mean(axis=1)
full_df['Avg_rental_value (€/m2 x month)'] = full_df[['Min_rental_value (€/m2 x␣
↪month)', 'Max_rental_value (€/m2 x month)']].mean(axis=1)
full_df.head()
```

[33]:

	Neighborhoods	Latitude	Longitude	Area Code		Housing_type	\
0	DUOMO	45.464585	9.189670	B	B12	Residential houses	
1	DUOMO	45.464585	9.189670	B	B12	Residential houses	
2	DUOMO	45.464585	9.189670	B	B12	Stately houses	
3	SAN BABILA	45.466521	9.197529	B	B12	Residential houses	
4	SAN BABILA	45.466521	9.197529	B	B12	Residential houses	

	Condition	Min_market_value (€/m2)	Max_market_value (€/m2)	\
0	Excellent	9000	12300	
1	Normal	7400	9000	
2	Excellent	11200	14300	
3	Excellent	9000	12300	
4	Normal	7400	9000	

	Min_rental_value (€/m2 x month)	Max_rental_value (€/m2 x month)	\
0	28.0	37.5	
1	23.0	28.0	
2	37.5	46.0	
3	28.0	37.5	
4	23.0	28.0	

	Avg_market_value (€/m2)	Avg_rental_value (€/m2 x month)
0	10650.0	32.75
1	8200.0	25.50
2	12750.0	41.75
3	10650.0	32.75
4	8200.0	25.50

2.13.2 Neighborhood distance from the city center

In order to predict prices as a function of distance (see below) I need to derive the distance to city center (Duomo di Milano is assigned as city center).

[34]:
```
Duomo_address='Duomo di Milano, Italy'
geolocater= Nominatim(user_agent="Center_search")
center= geolocater.geocode(Duomo_address)
lat_Duomo= center.latitude
lon_Duomo= center.longitude
print('The geograpical coordinate of {} are {}, {}.'.
 ↪format(Duomo_address,lat_Duomo, lon_Duomo))
```

```
The geograpical coordinate of Duomo di Milano, Italy are 45.46416835,
9.191621109614111.
```

We use the haver formula to calculate the distance between two spatial points.

[35]:
```
def haversine_vectorize(lon1, lat1, lon2, lat2):
    lon1, lat1, lon2, lat2 = map(np.radians, [lon1, lat1, lon2, lat2])

    newlon = lon2 - lon1
    newlat = lat2 - lat1

    haver_formula = np.sin(newlat/2.0)**2 + np.cos(lat1) * np.cos(lat2) * np.
 ↪sin(newlon/2.0)**2

    dist = 2 * np.arcsin(np.sqrt(haver_formula))
    km = 6367 * dist #6367 for distance in km, for miles use 3958
    return km

distance_center = haversine_vectorize(lat_Duomo, lon_Duomo,␣
 ↪full_df['Latitude'], full_df['Longitude'])
full_df = pd.concat([full_df, distance_center.rename('Distance_from_center␣
 ↪(km)')], axis=1)
full_df = full_df[full_df['Distance_from_center (km)']< 50]
full_df.head()
```

[35]:

	Neighborhoods	Latitude	Longitude	Area Code		Housing_type	\
0	DUOMO	45.464585	9.189670	B	B12	Residential houses	
1	DUOMO	45.464585	9.189670	B	B12	Residential houses	
2	DUOMO	45.464585	9.189670	B	B12	Stately houses	
3	SAN BABILA	45.466521	9.197529	B	B12	Residential houses	
4	SAN BABILA	45.466521	9.197529	B	B12	Residential houses	

	Condition	Min_market_value (€/m2)	Max_market_value (€/m2)	\
0	Excellent	9000	12300	
1	Normal	7400	9000	

```
2  Excellent                    11200                    14300
3  Excellent                     9000                    12300
4     Normal                     7400                     9000

   Min_rental_value (€/m2 x month)  Max_rental_value (€/m2 x month)  \
0                             28.0                             37.5
1                             23.0                             28.0
2                             37.5                             46.0
3                             28.0                             37.5
4                             23.0                             28.0

   Avg_market_value (€/m2)  Avg_rental_value (€/m2 x month)  \
0                  10650.0                            32.75
1                   8200.0                            25.50
2                  12750.0                            41.75
3                  10650.0                            32.75
4                   8200.0                            25.50

   Distance_from_center (km)
0                   0.221632
1                   0.221632
2                   0.221632
3                   0.705394
4                   0.705394
```

[36]:
```
def get_subcategory(original_df,housing_type,condition):
    '''
    this function slices the original df accoring to the house type and␣
↪conditions provided. Normalize the average prices
    in order to plot prices ranging between 0-1 and shows a bar plot subdivided␣
↪by area codes and a boxplot
    for each area

    Returns the sliced df
    '''
    # slice df based on conditions
    ex_res = original_df.loc[(original_df['Housing_type'] == housing_type) &␣
↪(full_df['Condition'] == condition)]
    ex_res = ex_res[['Area','Code','Avg_market_value (€/m2)','Avg_rental_value␣
↪(€/m2 x month)','Distance_from_center (km)']].reset_index(drop=True)
    if ex_res.shape[0] == 0:
        print('No conditions met. Exiting.')
        return
    # Normalize the single columns to better visualize the differences
    min_max_scaler = preprocessing.MinMaxScaler()
    x = ex_res[['Avg_market_value (€/m2)','Avg_rental_value (€/m2 x month)']].
↪values #returns a numpy array
```

```
    x_scaled = min_max_scaler.fit_transform(x)
    ex_res[['Norm_Avg_market_value (€/m2)','Norm_Avg_rental_value (€/m2 x␣
 ↪month)']] = pd.DataFrame(x_scaled)
    return ex_res

def my_bar_plot(ex_res,housing_type,condition):
    # BARPLOT OF different areas
    # Group the df
    grouped = ex_res.groupby(['Area','Code']).mean()


    fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(20,10), sharey=True)
#     plt.style.use('seaborn-paper')

    for (area, ax) in zip(grouped.index.get_level_values(0).unique(),axes.
 ↪flatten()):
        df= grouped.loc[area].reset_index()
        df.plot(x="Code", y=["Norm_Avg_market_value (€/m2)",␣
 ↪"Norm_Avg_rental_value (€/m2 x month)"], kind="bar",ax=ax)
        ax.get_legend().remove()
        if area == 'E':
            ax.legend(labels=['Norm BUY price','Norm RENT price'],loc='upper␣
 ↪right', fontsize=20)
        ax.set_xlabel('Area Code', color='black', fontsize=15, labelpad=13)
#         ax.set_ylabel('Average market value (€/m2)', color='black',␣
 ↪fontsize=20, labelpad=20)
        ax.tick_params(labelsize=15)

    fig.suptitle(f'Normalized prices of {housing_type} in {condition}␣
 ↪conditions', fontsize=25);


def my_boxplots(ex_res,housing_type,condition,buy_or_rent):
    if buy_or_rent == 'buy':
        ss = "Avg_market_value (€/m2)"
    else:
        ss = 'Avg_rental_value (€/m2 x month)'
    # BOXPLOT OF ALL AREAS
    fig1, axes = plt.subplots(figsize=(15,6))
#     st = plt.style.use('seaborn-paper')
    sns.boxplot(x="Area", y=ss, data=ex_res)
    axes.set_xlabel('Area Code', color='black', fontsize=20, labelpad=13)
    axes.set_ylabel(f'{ss}', color='black', fontsize=20, labelpad=20)
    axes.tick_params(labelsize=20)
```

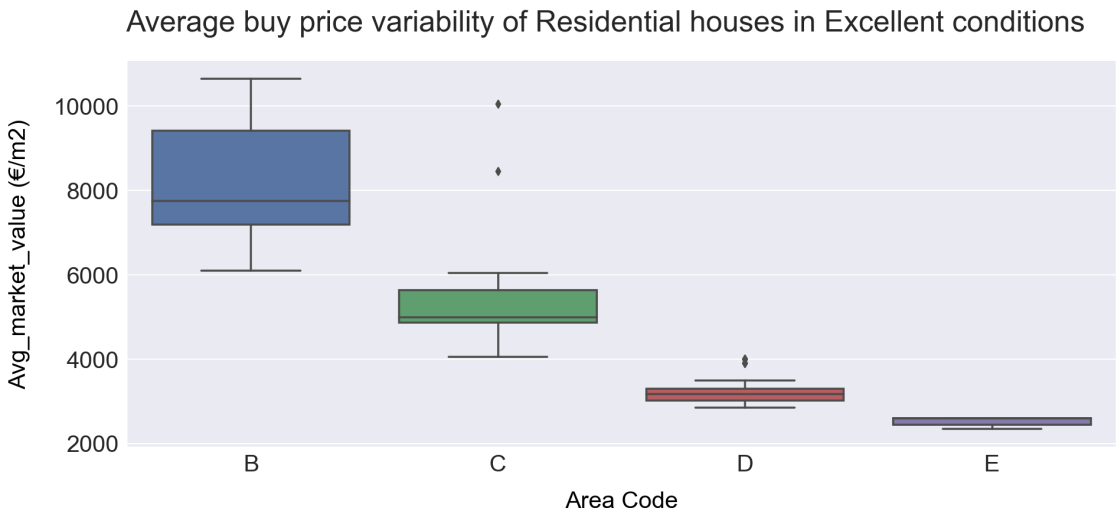
```
    fig1.suptitle(f'Average {buy_or_rent} price variability of {housing_type}␣
↪in {condition} conditions', fontsize=25);
```

2.13.3 Get only Excellent residential houses

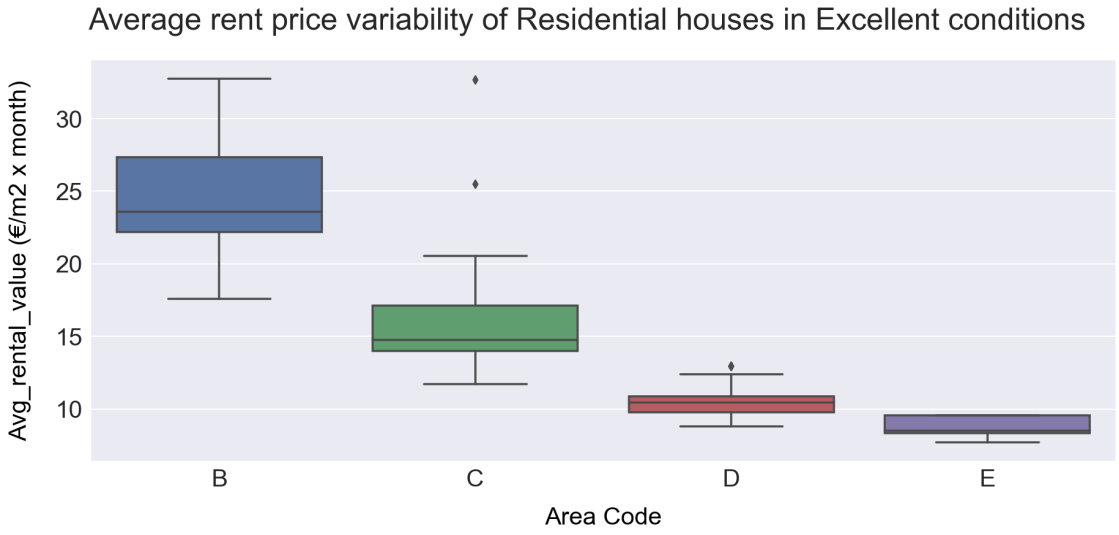
[37]:
```
res_ex = get_subcategory(full_df,'Residential houses','Excellent')
```

[38]:
```
mpl.style.use('seaborn')
my_bar_plot(res_ex,'Residential houses','Excellent')
```

[39]:
```
my_boxplots(res_ex,'Residential houses','Excellent','buy')
```



[40]:
```
my_boxplots(res_ex,'Residential houses','Excellent','rent')
```



2.13.4 Get only Normal residential houses

[41]:
```
res_nor = get_subcategory(full_df,'Residential houses','Normal')
my_bar_plot(res_nor,'Residential houses','Normal')
```

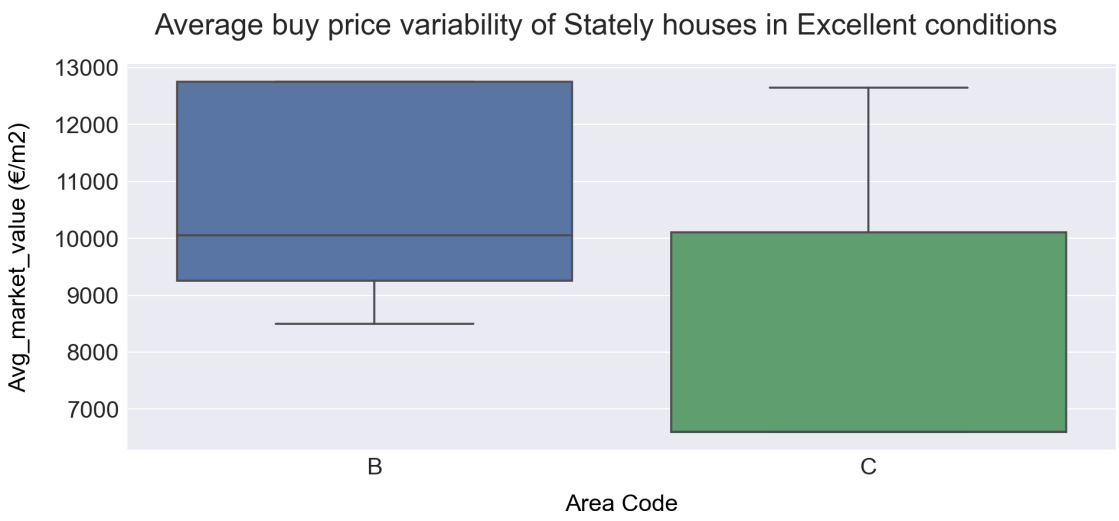
[42]:
```
my_boxplots(res_nor,'Residential houses','Normal','buy')
```

[43]:
```
my_boxplots(res_nor,'Residential houses','Normal','rent')
```

2.13.5 Get only Excellent stately houses

[44]:
```
sta_ex = get_subcategory(full_df,'Stately houses','Excellent')
my_bar_plot(sta_ex,'Stately houses','Excellent')
```

[45]:
```
my_boxplots(sta_ex,'Stately houses','Excellent','buy')
```



[46]:
```
my_boxplots(sta_ex,'Stately houses','Excellent','rent')
```

2.13.6 Get only Normal stately houses

[47]:
```
sta_nor = get_subcategory(full_df,'Stately houses','Normal')
```

```
No conditions met. Exiting.
```

2.13.7 Observations

Bar plots shown that the boroughs closer to the city center are more expensive; on the contrary, the ones farthest to the city center are the most affordable.

Let's try to predict the housing price based only on the distance to the city center.

2.14 3.7 Price prediction based on distance: Regression

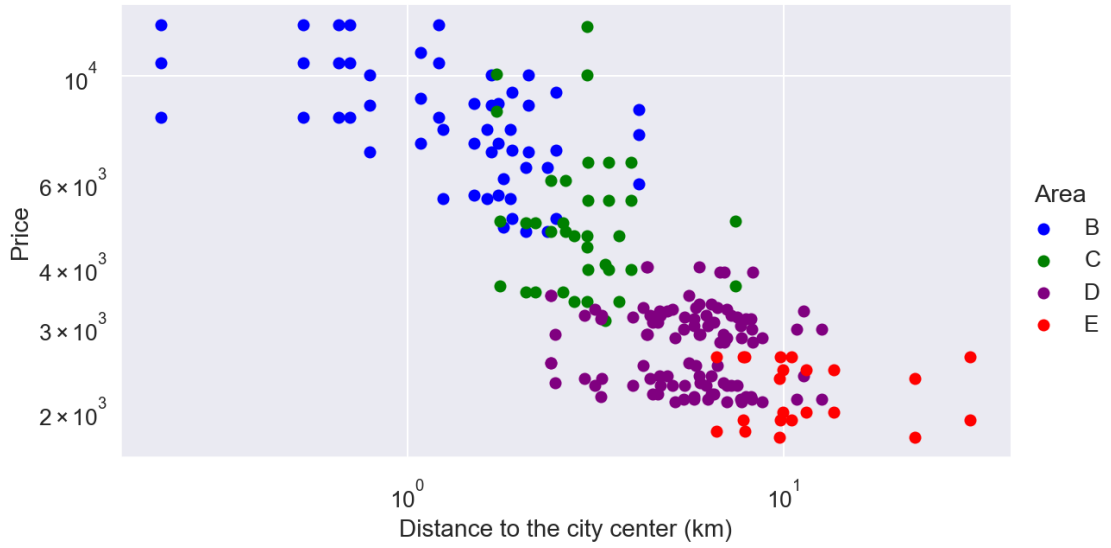
Box plots have shown that the most expensive residential houses are located in the B and C areas, and the most affordable ones are located in D and E areas, whereas not much of a difference emerged from stately houses which are also located only in B and C zones.

2.14.1 Create the scatter plots

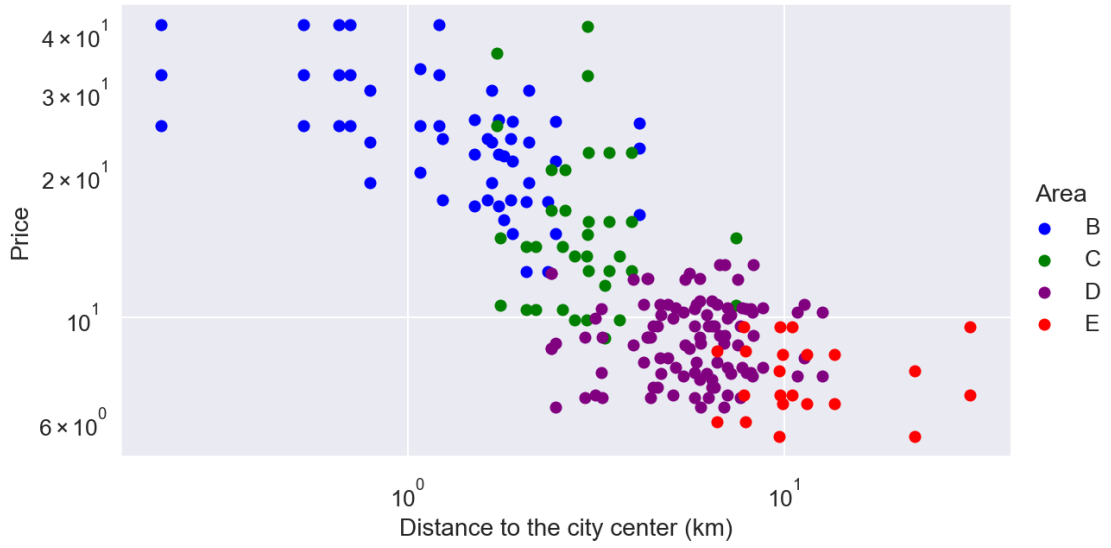
[48]:
```
def my_scatter(ex_res,buy_or_rent,scale):
    if buy_or_rent == 'buy':
        ss = 'Avg_market_value (€/m2)'
    elif buy_or_rent == 'rent':
        ss = 'Avg_rental_value (€/m2 x month)'

    d = {'color': ['blue', 'green', 'purple', 'red']}

    sns.set(font_scale = 1.2)
    fg = sns.FacetGrid(data=ex_res, hue='Area', hue_kws=d, height=5.0, aspect=1.
↪7)
    ttl = plt.title(f" ", weight='bold', size=13)

    plt.yscale(scale)
    plt.xscale(scale)
    fg.map(plt.scatter, 'Distance_from_center (km)', ss).add_legend()
    plt.xlabel('Distance to the city center (km)', fontsize=14)
    plt.ylabel('Price', fontsize=14)
```

Scatter plot using the full Dataset

[49]:
```
my_scatter(full_df,'buy','log')
```



[50]:
```
my_scatter(full_df,'rent','log')
```



As a linear correlation in log space is visible, lets fit a power law and an exponential model.

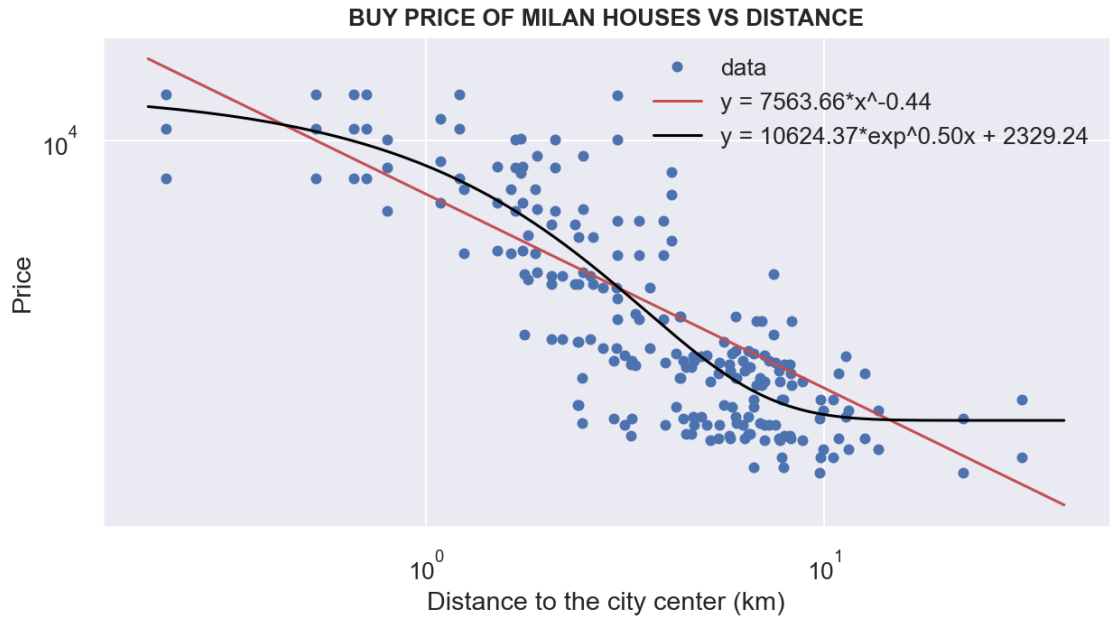
$y = a * x^b$

$y = c * exp^{-d*x} + e$

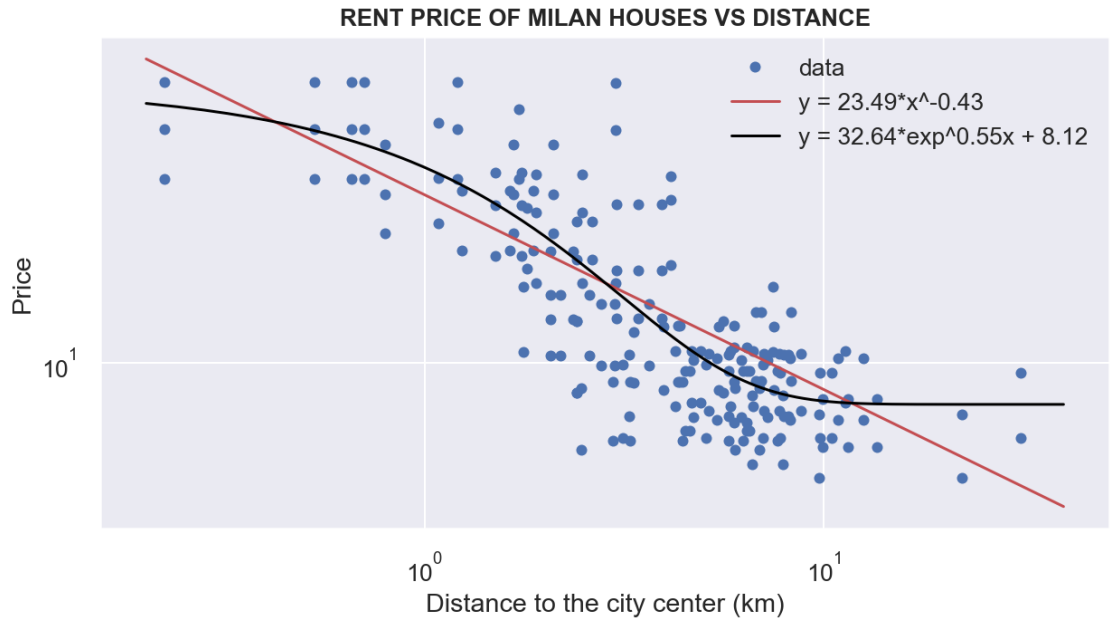
fit the model on the full_df without distinction of category

[51]:
```
def model_func(x, a, b):
    return a * x**b

def model_func2(x, c, d, e):
    return c * np.exp(-d*x) + e


def fit(df,buy_or_rent,scale):
    if buy_or_rent == 'buy':
        ss = 'Avg_market_value (€/m2)'
    elif buy_or_rent == 'rent':
        ss = 'Avg_rental_value (€/m2 x month)'

    x_er = df['Distance_from_center (km)']
    y_er = df[ss]

    p0 = (1.,-0.2) # starting point
    opt, pcov = curve_fit(model_func, x_er, y_er, p0)
    a, b = opt

    p0 = (1.,1.e-10,1.) # starting point
    opt, pcov = curve_fit(model_func2, x_er, y_er, p0)
    c, d, e = opt

    x2 = np.linspace(0.2, 40, 1000)
    y2 = model_func(x2, a, b)
    y3 = model_func2(x2, c, d, e)

    # plot
    fig, ax = plt.subplots(figsize=(10,5))

    plt.rcParams.update({'font.size': 20})
    ax.plot(x_er, y_er, 'bo', label='data')
    ax.plot(x2, y2, color='r', label=f'y = {a:.2f}*x^{b:.2f}')
    ax.plot(x2, y3, color='black', label=f'y = {c:.2f}*exp^{d:.2f}x + {e:.2f}')
    plt.yscale(scale)
    plt.xscale(scale)

    ax.legend()
    plt.title(f"{buy_or_rent.upper()} PRICE OF MILAN HOUSES VS DISTANCE",␣
↪weight='bold', size=13)
    plt.xlabel('Distance to the city center (km)')
    plt.ylabel('Price')
```

[52]:
```
fit(full_df,'buy','log')
```



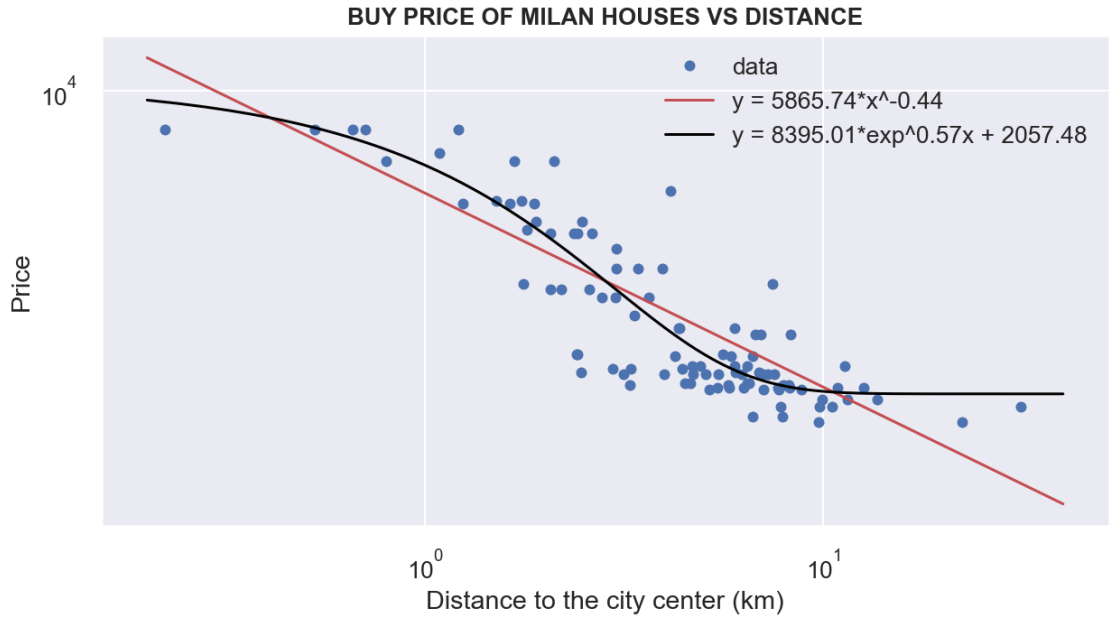
[53]:
```
fit(full_df,'rent','log')
```



The same analysis can be performed on residential houses (normal and/or excellent conditions) and on stately houses. Let's see how normal residential houses will behave.

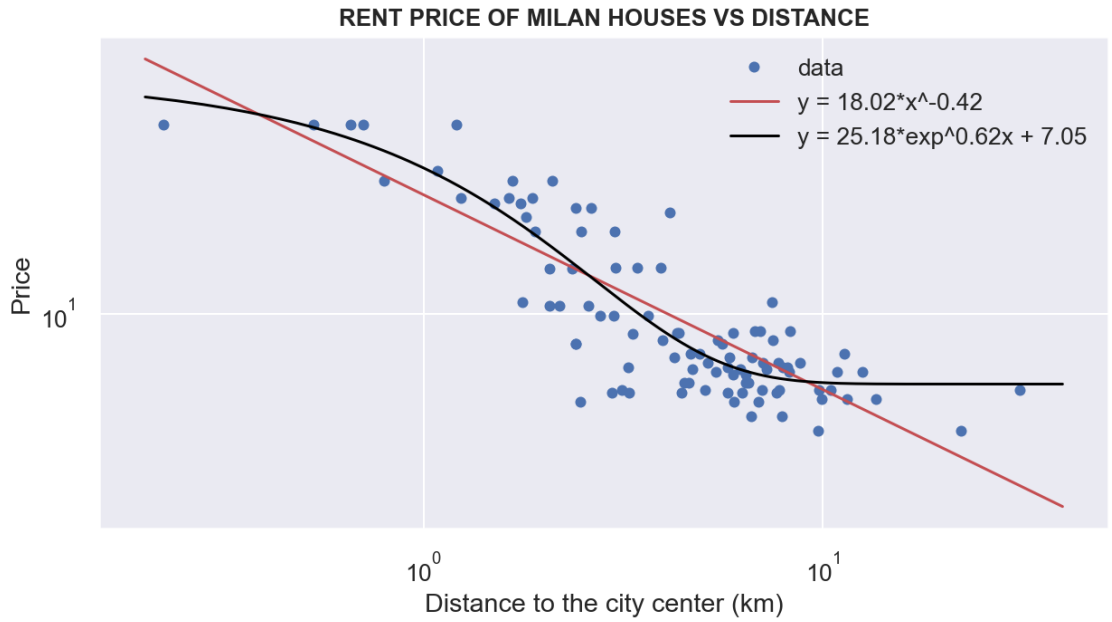
Buy a Residential house in Normal conditions

[54]:
```
fit(res_nor,'buy','log')
```



Rent a Residential house in Normal conditions

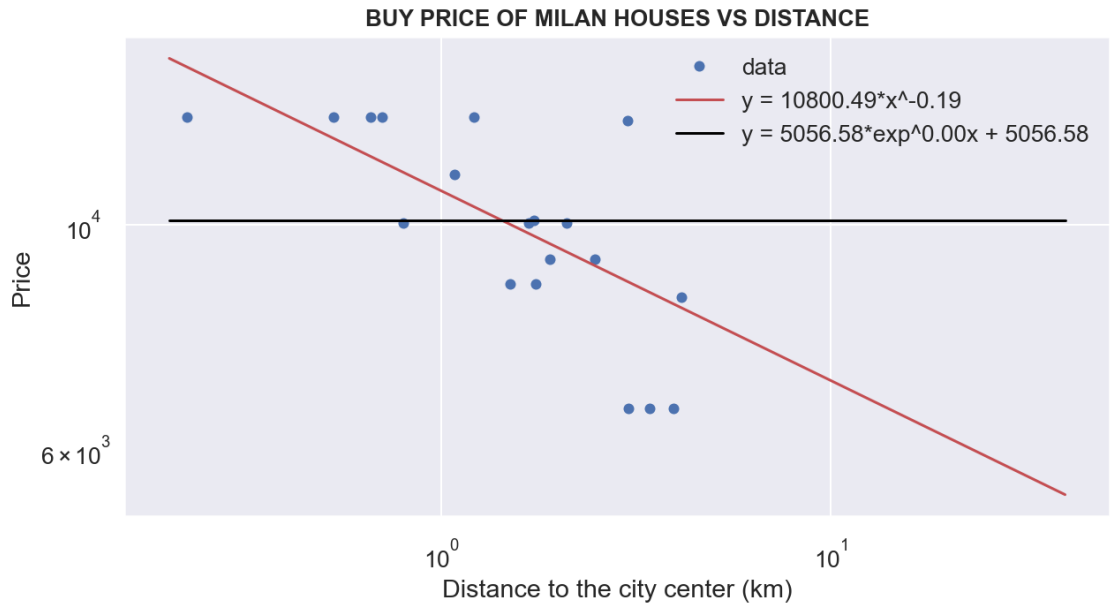
[55]:
```
fit(res_nor,'rent','log')
```



Buy a Stately house in Milan

[56]:
```
fit(sta_ex,'buy','log')
```

```
/Users/segatto/opt/anaconda3/envs/Data_Science_env/lib/python3.6/site-
packages/scipy/optimize/minpack.py:829: OptimizeWarning: Covariance of the
parameters could not be estimated
  category=OptimizeWarning)
```



The exponential model is overfitted for this category. Use a power-law model.

2.15 3.8 Venues analysis: Neighborhood segmentation and clustering

Objective: segment all Milan' neighborhoods and cluster them in order to maximize dissimilarity among different clusters and minimize similarity among neighborhoods within the same cluseter. To do so, I use an unsupervised learning algorithm such as K-Means and I explicitly inclue venues in the full_df in order to consider, along with price, all services provided within 1 km, i.e. venues.

[57]:
```
milan_venues.head()
```

[57]:

	Neighborhoods	Latitude	Longitude	Venue_Name	\
0	DUOMO	45.464585	9.18967	Piazza del Duomo	
1	DUOMO	45.464585	9.18967	Galleria Vittorio Emanuele II	
2	DUOMO	45.464585	9.18967	Room Mate Giulia Hotel	
3	DUOMO	45.464585	9.18967	Terrazze del Duomo	
4	DUOMO	45.464585	9.18967	Palazzo Reale	

	Venue_Latitude	Venue_Longitude	Venue_Category

```
0       45.464190         9.189527                Plaza
1       45.465577         9.190024  Monument / Landmark
2       45.465250         9.189396                Hotel
3       45.464207         9.191075       Scenic Lookout
4       45.462960         9.191348          Art Gallery
```

[58]:
```
full_df.head()
```

[58]:
```
  Neighborhoods   Latitude  Longitude Area Code        Housing_type  \
0         DUOMO  45.464585   9.189670    B  B12  Residential houses
1         DUOMO  45.464585   9.189670    B  B12  Residential houses
2         DUOMO  45.464585   9.189670    B  B12      Stately houses
3    SAN BABILA  45.466521   9.197529    B  B12  Residential houses
4    SAN BABILA  45.466521   9.197529    B  B12  Residential houses

   Condition  Min_market_value (€/m2)  Max_market_value (€/m2)  \
0  Excellent                     9000                    12300
1     Normal                     7400                     9000
2  Excellent                    11200                    14300
3  Excellent                     9000                    12300
4     Normal                     7400                     9000

   Min_rental_value (€/m2 x month)  Max_rental_value (€/m2 x month)  \
0                             28.0                             37.5
1                             23.0                             28.0
2                             37.5                             46.0
3                             28.0                             37.5
4                             23.0                             28.0

   Avg_market_value (€/m2)  Avg_rental_value (€/m2 x month)  \
0                  10650.0                            32.75
1                   8200.0                            25.50
2                  12750.0                            41.75
3                  10650.0                            32.75
4                   8200.0                            25.50

   Distance_from_center (km)
0                   0.221632
1                   0.221632
2                   0.221632
3                   0.705394
4                   0.705394
```

Add the average house price independently on the house type and condition

[59]:
```
df_price = full_df[['Code','Avg_market_value (€/m2)','Avg_rental_value (€/m2 x␣
↪month)']]
```

```
avg_price = df_price.groupby(by='Code').mean()
avg_price.rename(columns={'Avg_market_value (€/m2)':
 ↪'Buy_price','Avg_rental_value (€/m2 x month)':'Rent_price'},inplace=True)
avg_price.reset_index(inplace=True)
avg_price.head()
```

[59]:

	Code	Buy_price	Rent_price
0	B12	10533.333333	33.333333
1	B13	7333.333333	21.750000
2	B15	9150.000000	26.566667
3	B16	8566.666667	24.433333
4	B17	7100.000000	20.866667

[60]:
```
full_df_venues = pd.merge(milan_venues,full_df[['Neighborhoods','Code']],␣
 ↪on='Neighborhoods')
df_to_segment = pd.merge(avg_price,full_df_venues,on='Code')
df_to_segment = df_to_segment.drop_duplicates()

# df_to_segment = df_to_segment[[df_to_segment.columns[-1]] +␣
 ↪list(df_to_segment.columns[:-1])]
df_to_segment.head()
```

[60]:

	Code	Buy_price	Rent_price	Neighborhoods	Latitude	Longitude	\
0	B12	10533.333333	33.333333	DUOMO	45.464585	9.18967	
3	B12	10533.333333	33.333333	DUOMO	45.464585	9.18967	
6	B12	10533.333333	33.333333	DUOMO	45.464585	9.18967	
9	B12	10533.333333	33.333333	DUOMO	45.464585	9.18967	
12	B12	10533.333333	33.333333	DUOMO	45.464585	9.18967	

	Venue_Name	Venue_Latitude	Venue_Longitude	\
0	Piazza del Duomo	45.464190	9.189527	
3	Galleria Vittorio Emanuele II	45.465577	9.190024	
6	Room Mate Giulia Hotel	45.465250	9.189396	
9	Terrazze del Duomo	45.464207	9.191075	
12	Palazzo Reale	45.462960	9.191348	

	Venue_Category
0	Plaza
3	Monument / Landmark
6	Hotel
9	Scenic Lookout
12	Art Gallery

[61]:
```
df_to_segment.shape
```

[61]: (6616, 10)

2.15.1 Number of venues for each neighborhood

[62]:
```
df_to_segment.groupby('Neighborhoods').count()['Venue_Category'].head(30)
```

[62]:
```
Neighborhoods
AFFORI                 38
ARCO DELLA PACE       100
ARGONNE               100
BAGGIO                 15
BARONA                 13
BAUSAN                 62
BAZZI                 100
BICOCCA                79
BIGNAMI                58
BISCEGLIE              15
BONOLA                 28
BOVISA                 55
BOVISASCA              29
BRERA                 100
CADORNA               100
CAIROLI               100
CAPRILLI               55
CASCINA MERLATA        19
CENISIO               100
CERTOSA                44
CHIESA ROSSA           36
CITTÀ STUDI           100
CITY LIFE             100
COMASINA               26
CORSICA               100
CORSO BUENOS AIRES    100
CRESCENZAGO            28
DUOMO                 100
EXPO                   18
FAMAGOSTA              53
Name: Venue_Category, dtype: int64
```

[63]:
```
print('There are {} uniques categories.'
 ↪format(len(df_to_segment['Venue_Category'].unique())))
```

```
There are 296 uniques categories.
```

[64]:
```
df_to_segment['Venue_Category'].unique()[:50]
```

[64]:
```
array(['Plaza', 'Monument / Landmark', 'Hotel', 'Scenic Lookout',
       'Art Gallery', 'Ice Cream Shop', 'Pastry Shop', 'Bakery', 'Café',
       'Kitchen Supply Store', 'Art Museum', 'Coffee Shop',
```

```
       'Department Store', 'Opera House', 'Boutique', 'Wine Bar',
       'Electronics Store', 'Korean Restaurant', 'Gourmet Shop',
       'Italian Restaurant', 'Bookstore', 'Lounge', 'Food & Drink Shop',
       'Pizza Place', 'Sporting Goods Shop', 'Bistro',
       'Salon / Barbershop', 'Gift Shop', 'Cosmetics Shop',
       'Dessert Shop', 'Cocktail Bar', "Men's Store",
       'Japanese Restaurant', 'Church', 'Bar', 'Bike Shop', 'Road',
       "Women's Store", 'Toy / Game Store', 'Accessories Store',
       'Perfume Shop', 'Jewelry Store', 'Restaurant', 'Camera Store',
       'Chocolate Shop', 'Mediterranean Restaurant', 'Burger Joint',
       'Museum', 'Furniture / Home Store', 'Music School'], dtype=object)
```

2.15.2 Analyze each neighborhoods

[65]:
```
# one hot encoding
Milan_onehot = pd.get_dummies(df_to_segment[['Venue_Category']], prefix="",␣
 ↪prefix_sep="")

# add borough code and neighborhood column back to dataframe
Milan_onehot['Borough code'] = df_to_segment['Code']
Milan_onehot['Neighborhoods'] = df_to_segment['Neighborhoods']
Milan_onehot['Latitude'] = df_to_segment['Latitude']
Milan_onehot['Longitude'] = df_to_segment['Longitude']
Milan_onehot['Rent_price'] = df_to_segment['Rent_price']
Milan_onehot['Buy_price'] = df_to_segment['Buy_price']
# move borough code and neighborhood columsn to the first column
fixed_columns = list(Milan_onehot.columns[-6:]) + list(Milan_onehot.columns[:
 ↪-6])
Milan_onehot = Milan_onehot[fixed_columns]

Milan_onehot.head()
```

[65]:

	Borough code	Neighborhoods	Latitude	Longitude	Rent_price	Buy_price	\
0	B12	DUOMO	45.464585	9.18967	33.333333	10533.333333	
3	B12	DUOMO	45.464585	9.18967	33.333333	10533.333333	
6	B12	DUOMO	45.464585	9.18967	33.333333	10533.333333	
9	B12	DUOMO	45.464585	9.18967	33.333333	10533.333333	
12	B12	DUOMO	45.464585	9.18967	33.333333	10533.333333	

	Abruzzo Restaurant	Accessories Store	Adult Education Center	\
0	0	0	0	
3	0	0	0	
6	0	0	0	
9	0	0	0	
12	0	0	0	

```
    African Restaurant  Agriturismo  Airport  American Restaurant  \
0                    0            0        0                    0
3                    0            0        0                    0
6                    0            0        0                    0
9                    0            0        0                    0
12                   0            0        0                    0

    Amphitheater  Arcade  Argentinian Restaurant  Art Gallery  Art Museum  \
0              0       0                       0            0           0
3              0       0                       0            0           0
6              0       0                       0            0           0
9              0       0                       0            0           0
12             0       0                       0            1           0

    Arts & Crafts Store  Arts & Entertainment  Asian Restaurant  \
0                     0                     0                 0
3                     0                     0                 0
6                     0                     0                 0
9                     0                     0                 0
12                    0                     0                 0

    Athletics & Sports  Auto Garage  BBQ Joint  Bagel Shop  Bakery  Ballroom  \
0                    0            0          0           0       0         0
3                    0            0          0           0       0         0
6                    0            0          0           0       0         0
9                    0            0          0           0       0         0
12                   0            0          0           0       0         0

    Bar  Basketball Court  Bed & Breakfast  Beer Bar  Beer Garden  Beer Store  \
0     0                 0                0         0            0           0
3     0                 0                0         0            0           0
6     0                 0                0         0            0           0
9     0                 0                0         0            0           0
12    0                 0                0         0            0           0

    Betting Shop  Bike Shop  Bistro  Board Shop  Boarding House  Bookstore  \
0              0          0       0           0               0          0
3              0          0       0           0               0          0
6              0          0       0           0               0          0
9              0          0       0           0               0          0
12             0          0       0           0               0          0

    Boutique  Bowling Alley  Brazilian Restaurant  Breakfast Spot  Brewery  \
0          0              0                     0               0        0
3          0              0                     0               0        0
6          0              0                     0               0        0
```

```
9          0              0                    0               0        0
12         0              0                    0               0        0

    Bubble Tea Shop  Buffet  Burger Joint  Bus Station  Bus Stop  \
0                 0       0             0            0         0
3                 0       0             0            0         0
6                 0       0             0            0         0
9                 0       0             0            0         0
12                0       0             0            0         0

    Business Service  Butcher  Cafeteria  Café  Camera Store  \
0                  0        0          0     0             0
3                  0        0          0     0             0
6                  0        0          0     0             0
9                  0        0          0     0             0
12                 0        0          0     0             0

    Campanian Restaurant  Campground  Canal  Candy Store  \
0                      0           0      0            0
3                      0           0      0            0
6                      0           0      0            0
9                      0           0      0            0
12                     0           0      0            0

    Caribbean Restaurant  Castle  Cemetery  Cheese Shop  Chinese Restaurant  \
0                      0       0         0            0                   0
3                      0       0         0            0                   0
6                      0       0         0            0                   0
9                      0       0         0            0                   0
12                     0       0         0            0                   0

    Chocolate Shop  Church  City  Climbing Gym  Clothing Store  Cocktail Bar  \
0                0       0     0             0               0             0
3                0       0     0             0               0             0
6                0       0     0             0               0             0
9                0       0     0             0               0             0
12               0       0     0             0               0             0

    Coffee Shop  College Arts Building  College Cafeteria  Comedy Club  \
0             0                      0                  0            0
3             0                      0                  0            0
6             0                      0                  0            0
9             0                      0                  0            0
12            0                      0                  0            0

    Comic Shop  Convenience Store  Convention Center  Cosmetics Shop  \
0            0                  0                  0               0
```

```
3            0                 0                 0              0
6            0                 0                 0              0
9            0                 0                 0              0
12           0                 0                 0              0

    Coworking Space  Creperie  Cultural Center  Cupcake Shop  Dance Studio  \
0                 0         0                0             0             0
3                 0         0                0             0             0
6                 0         0                0             0             0
9                 0         0                0             0             0
12                0         0                0             0             0

    Deli / Bodega  Department Store  Design Studio  Dessert Shop  \
0               0                 0              0             0
3               0                 0              0             0
6               0                 0              0             0
9               0                 0              0             0
12              0                 0              0             0

    Dim Sum Restaurant  Diner  Discount Store  Dive Bar  Doner Restaurant  \
0                    0      0               0         0                 0
3                    0      0               0         0                 0
6                    0      0               0         0                 0
9                    0      0               0         0                 0
12                   0      0               0         0                 0

    Electronics Store  Empanada Restaurant  Event Space  Exhibit  Fabric Shop  \
0                   0                    0            0        0            0
3                   0                    0            0        0            0
6                   0                    0            0        0            0
9                   0                    0            0        0            0
12                  0                    0            0        0            0

    Falafel Restaurant  Farm  Farmers Market  Fast Food Restaurant  \
0                    0     0               0                     0
3                    0     0               0                     0
6                    0     0               0                     0
9                    0     0               0                     0
12                   0     0               0                     0

    Filipino Restaurant  Film Studio  Fish & Chips Shop  Fish Market  \
0                     0            0                  0            0
3                     0            0                  0            0
6                     0            0                  0            0
9                     0            0                  0            0
12                    0            0                  0            0
```

```
    Flea Market  Flower Shop  Food  Food & Drink Shop  Food Court  Food Truck  \
0             0            0     0                  0           0           0
3             0            0     0                  0           0           0
6             0            0     0                  0           0           0
9             0            0     0                  0           0           0
12            0            0     0                  0           0           0

    Football Stadium  Fountain  French Restaurant  Fried Chicken Joint  \
0                  0         0                  0                    0
3                  0         0                  0                    0
6                  0         0                  0                    0
9                  0         0                  0                    0
12                 0         0                  0                    0

    Friterie  Frozen Yogurt Shop  Furniture / Home Store  Gaming Cafe  Garden  \
0          0                   0                       0            0       0
3          0                   0                       0            0       0
6          0                   0                       0            0       0
9          0                   0                       0            0       0
12         0                   0                       0            0       0

    Garden Center  Gas Station  Gastropub  Gay Bar  General Entertainment  \
0               0            0          0        0                      0
3               0            0          0        0                      0
6               0            0          0        0                      0
9               0            0          0        0                      0
12              0            0          0        0                      0

    German Restaurant  Gift Shop  Golf Course  Gourmet Shop  \
0                   0          0            0             0
3                   0          0            0             0
6                   0          0            0             0
9                   0          0            0             0
12                  0          0            0             0

    Government Building  Greek Restaurant  Grocery Store  Gym  \
0                     0                 0              0    0
3                     0                 0              0    0
6                     0                 0              0    0
9                     0                 0              0    0
12                    0                 0              0    0

    Gym / Fitness Center  Gym Pool  Harbor / Marina  Hardware Store  \
0                      0         0                0               0
3                      0         0                0               0
6                      0         0                0               0
9                      0         0                0               0
```

```
12                    0         0               0              0

    Hawaiian Restaurant  Health Food Store  Historic Site  History Museum  \
0                     0                  0              0               0
3                     0                  0              0               0
6                     0                  0              0               0
9                     0                  0              0               0
12                    0                  0              0               0

    Hobby Shop  Hockey Arena  Hostel  Hotel  Hotel Bar  IT Services  \
0            0             0       0      0          0            0
3            0             0       0      0          0            0
6            0             0       0      1          0            0
9            0             0       0      0          0            0
12           0             0       0      0          0            0

    Ice Cream Shop  Indian Restaurant  Indie Movie Theater  Indie Theater  \
0                0                  0                    0              0
3                0                  0                    0              0
6                0                  0                    0              0
9                0                  0                    0              0
12               0                  0                    0              0

    Intersection  Irish Pub  Italian Restaurant  Japanese Restaurant  \
0              0          0                   0                    0
3              0          0                   0                    0
6              0          0                   0                    0
9              0          0                   0                    0
12             0          0                   0                    0

    Jazz Club  Jewelry Store  Juice Bar  Karaoke Bar  Kebab Restaurant  \
0           0              0          0            0                 0
3           0              0          0            0                 0
6           0              0          0            0                 0
9           0              0          0            0                 0
12          0              0          0            0                 0

    Kids Store  Kitchen Supply Store  Korean Restaurant  \
0            0                     0                  0
3            0                     0                  0
6            0                     0                  0
9            0                     0                  0
12           0                     0                  0

    Latin American Restaurant  Lebanese Restaurant  Lombard Restaurant  \
0                           0                    0                   0
3                           0                    0                   0
```

```
6                          0                  0                  0
9                          0                  0                  0
12                         0                  0                  0

    Lounge  Market  Martial Arts School  Mattress Store  Medical Center  \
0        0       0                    0               0               0
3        0       0                    0               0               0
6        0       0                    0               0               0
9        0       0                    0               0               0
12       0       0                    0               0               0

    Mediterranean Restaurant  Men's Store  Metro Station  Mexican Restaurant  \
0                          0            0              0                   0
3                          0            0              0                   0
6                          0            0              0                   0
9                          0            0              0                   0
12                         0            0              0                   0

    Middle Eastern Restaurant  Miscellaneous Shop  Mobile Phone Shop  \
0                           0                   0                  0
3                           0                   0                  0
6                           0                   0                  0
9                           0                   0                  0
12                          0                   0                  0

    Mongolian Restaurant  Monument / Landmark  Moroccan Restaurant  \
0                      0                    0                    0
3                      0                    1                    0
6                      0                    0                    0
9                      0                    0                    0
12                     0                    0                    0

    Motorcycle Shop  Movie Theater  Multiplex  Museum  Music School  \
0                 0              0          0       0             0
3                 0              0          0       0             0
6                 0              0          0       0             0
9                 0              0          0       0             0
12                0              0          0       0             0

    Music Store  Music Venue  Neighborhood  Nightclub  Noodle House  Office  \
0             0            0             0          0             0       0
3             0            0             0          0             0       0
6             0            0             0          0             0       0
9             0            0             0          0             0       0
12            0            0             0          0             0       0

    Opera House  Optical Shop  Other Nightlife  Outdoor Sculpture  \
```

```
0             0             0               0                 0
3             0             0               0                 0
6             0             0               0                 0
9             0             0               0                 0
12            0             0               0                 0

    Outdoors & Recreation  Outlet Mall  Outlet Store  Palace  \
0                       0            0             0       0
3                       0            0             0       0
6                       0            0             0       0
9                       0            0             0       0
12                      0            0             0       0

    Paper / Office Supplies Store  Park  Pastry Shop  Pedestrian Plaza  \
0                               0     0            0                 0
3                               0     0            0                 0
6                               0     0            0                 0
9                               0     0            0                 0
12                              0     0            0                 0

    Performing Arts Venue  Perfume Shop  Persian Restaurant  Pet Store  \
0                       0             0                   0          0
3                       0             0                   0          0
6                       0             0                   0          0
9                       0             0                   0          0
12                      0             0                   0          0

    Pharmacy  Photography Lab  Piadineria  Pizza Place  Planetarium  Platform  \
0          0                0           0            0            0         0
3          0                0           0            0            0         0
6          0                0           0            0            0         0
9          0                0           0            0            0         0
12         0                0           0            0            0         0

    Playground  Plaza  Pool  Pool Hall  Print Shop  Pub  Public Art  \
0            0      1     0          0           0    0           0
3            0      0     0          0           0    0           0
6            0      0     0          0           0    0           0
9            0      0     0          0           0    0           0
12           0      0     0          0           0    0           0

    Puglia Restaurant  Racecourse  Racetrack  Radio Station  Ramen Restaurant  \
0                   0           0          0              0                 0
3                   0           0          0              0                 0
6                   0           0          0              0                 0
9                   0           0          0              0                 0
12                  0           0          0              0                 0
```

```
    Record Shop  Recording Studio  Recreation Center  \
0             0                 0                  0
3             0                 0                  0
6             0                 0                  0
9             0                 0                  0
12            0                 0                  0

    Residential Building (Apartment / Condo)  Rest Area  Restaurant  River  \
0                                          0          0           0      0
3                                          0          0           0      0
6                                          0          0           0      0
9                                          0          0           0      0
12                                         0          0           0      0

    Road  Rock Club  Roman Restaurant  Russian Restaurant  Salad Place  \
0      0          0                 0                   0            0
3      0          0                 0                   0            0
6      0          0                 0                   0            0
9      0          0                 0                   0            0
12     0          0                 0                   0            0

    Salon / Barbershop  Sandwich Place  Sardinian Restaurant  \
0                    0              0                     0
3                    0              0                     0
6                    0              0                     0
9                    0              0                     0
12                   0              0                     0

    Sauna / Steam Room  Scenic Lookout  Science Museum  Seafood Restaurant  \
0                    0              0               0                   0
3                    0              0               0                   0
6                    0              0               0                   0
9                    0              1               0                   0
12                   0              0               0                   0

    Shipping Store  Shoe Store  Shopping Mall  Shopping Plaza  \
0                0           0              0               0
3                0           0              0               0
6                0           0              0               0
9                0           0              0               0
12               0           0              0               0

    Sicilian Restaurant  Skate Park  Smoke Shop  Snack Place  Soccer Field  \
0                     0           0           0            0             0
3                     0           0           0            0             0
6                     0           0           0            0             0
```

```
9                    0           0           0            0             0
12                   0           0           0            0             0

    Soccer Stadium  Social Club  South American Restaurant  \
0                0            0                          0
3                0            0                          0
6                0            0                          0
9                0            0                          0
12               0            0                          0

    South Tyrolean Restaurant  Spa  Spanish Restaurant  Speakeasy  \
0                           0    0                   0          0
3                           0    0                   0          0
6                           0    0                   0          0
9                           0    0                   0          0
12                          0    0                   0          0

    Sporting Goods Shop  Sports Bar  Sports Club  Sri Lankan Restaurant  \
0                     0           0            0                      0
3                     0           0            0                      0
6                     0           0            0                      0
9                     0           0            0                      0
12                    0           0            0                      0

    Stadium  Stationery Store  Steakhouse  Street Art  Strip Club  \
0         0                 0           0           0           0
3         0                 0           0           0           0
6         0                 0           0           0           0
9         0                 0           0           0           0
12        0                 0           0           0           0

    Supermarket  Sushi Restaurant  Szechuan Restaurant  Tapas Restaurant  \
0             0                 0                    0                 0
3             0                 0                    0                 0
6             0                 0                    0                 0
9             0                 0                    0                 0
12            0                 0                    0                 0

    Tea Room  Tennis Court  Tennis Stadium  Thai Restaurant  Theater  \
0          0             0               0                0        0
3          0             0               0                0        0
6          0             0               0                0        0
9          0             0               0                0        0
12         0             0               0                0        0

    Theme Park Ride / Attraction  Thrift / Vintage Store  Toy / Game Store  \
0                              0                       0                 0
```

```
3                              0                      0                 0
6                              0                      0                 0
9                              0                      0                 0
12                             0                      0                 0

    Track  Trail  Train  Train Station  Tram Station  Trattoria/Osteria  \
0       0      0      0              0             0                  0
3       0      0      0              0             0                  0
6       0      0      0              0             0                  0
9       0      0      0              0             0                  0
12      0      0      0              0             0                  0

    Tunnel  Turkish Restaurant  Tuscan Restaurant  \
0        0                   0                  0
3        0                   0                  0
6        0                   0                  0
9        0                   0                  0
12       0                   0                  0

    Vegetarian / Vegan Restaurant  Video Game Store  Video Store  \
0                               0                 0            0
3                               0                 0            0
6                               0                 0            0
9                               0                 0            0
12                              0                 0            0

    Vietnamese Restaurant  Volleyball Court  Warehouse Store  Water Park  \
0                       0                 0                0           0
3                       0                 0                0           0
6                       0                 0                0           0
9                       0                 0                0           0
12                      0                 0                0           0

    Wine Bar  Wine Shop  Winery  Women's Store  Yoga Studio
0          0          0       0              0            0
3          0          0       0              0            0
6          0          0       0              0            0
9          0          0       0              0            0
12         0          0       0              0            0
```

2.15.3 Group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

[66]:
```
Milan_grouped = Milan_onehot.groupby(["Borough code", "Neighborhoods",␣
↪"Latitude", "Longitude","Rent_price",'Buy_price']).mean().reset_index()

print(Milan_grouped.shape)
Milan_grouped.head()
```

```
(98, 302)
```

[66]:

	Borough code	Neighborhoods	Latitude	Longitude	Rent_price	\
0	B12	CAIROLI	45.468701	9.181697	33.333333	
1	B12	DUOMO	45.464585	9.189670	33.333333	
2	B12	MISSORI	45.459828	9.189555	33.333333	
3	B12	MONTENAPOLEONE	45.470015	9.192868	33.333333	
4	B12	SAN BABILA	45.466521	9.197529	33.333333	

	Buy_price	Abruzzo Restaurant	Accessories Store	\
0	10533.333333	0.0	0.010000	
1	10533.333333	0.0	0.010000	
2	10533.333333	0.0	0.000000	
3	10533.333333	0.0	0.020000	
4	10533.333333	0.0	0.010101	

	Adult Education Center	African Restaurant	Agriturismo	Airport	\
0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	

	American Restaurant	Amphitheater	Arcade	Argentinian Restaurant	\
0	0.0	0.0	0.0	0.00	
1	0.0	0.0	0.0	0.00	
2	0.0	0.0	0.0	0.01	
3	0.0	0.0	0.0	0.00	
4	0.0	0.0	0.0	0.00	

	Art Gallery	Art Museum	Arts & Crafts Store	Arts & Entertainment	\
0	0.020000	0.030000	0.0	0.0	
1	0.020000	0.030000	0.0	0.0	
2	0.020000	0.020000	0.0	0.0	
3	0.020000	0.020000	0.0	0.0	
4	0.030303	0.010101	0.0	0.0	

	Asian Restaurant	Athletics & Sports	Auto Garage	BBQ Joint	Bagel Shop	\

```
0          0.000000                0.01          0.0   0.000000         0.0
1          0.000000                0.00          0.0   0.000000         0.0
2          0.000000                0.00          0.0   0.000000         0.0
3          0.000000                0.00          0.0   0.000000         0.0
4          0.010101                0.00          0.0   0.010101         0.0

     Bakery  Ballroom       Bar  Basketball Court  Bed & Breakfast  Beer Bar  \
0  0.020000       0.0  0.000000               0.0              0.0       0.0
1  0.030000       0.0  0.020000               0.0              0.0       0.0
2  0.020000       0.0  0.020000               0.0              0.0       0.0
3  0.030000       0.0  0.010000               0.0              0.0       0.0
4  0.030303       0.0  0.010101               0.0              0.0       0.0

   Beer Garden  Beer Store  Betting Shop  Bike Shop    Bistro  Board Shop  \
0          0.0         0.0           0.0       0.00  0.000000         0.0
1          0.0         0.0           0.0       0.01  0.010000         0.0
2          0.0         0.0           0.0       0.01  0.020000         0.0
3          0.0         0.0           0.0       0.00  0.010000         0.0
4          0.0         0.0           0.0       0.00  0.020202         0.0

   Boarding House  Bookstore  Boutique  Bowling Alley  Brazilian Restaurant  \
0             0.0   0.000000  0.010000            0.0                   0.0
1             0.0   0.010000  0.130000            0.0                   0.0
2             0.0   0.020000  0.040000            0.0                   0.0
3             0.0   0.020000  0.170000            0.0                   0.0
4             0.0   0.010101  0.050505            0.0                   0.0

   Breakfast Spot  Brewery  Bubble Tea Shop  Buffet  Burger Joint  \
0             0.0      0.0              0.0     0.0      0.000000
1             0.0      0.0              0.0     0.0      0.010000
2             0.0      0.0              0.0     0.0      0.010000
3             0.0      0.0              0.0     0.0      0.000000
4             0.0      0.0              0.0     0.0      0.010101

   Bus Station  Bus Stop  Business Service  Butcher  Cafeteria      Café  \
0          0.0       0.0               0.0      0.0        0.0  0.030000
1          0.0       0.0               0.0      0.0        0.0  0.020000
2          0.0       0.0               0.0      0.0        0.0  0.050000
3          0.0       0.0               0.0      0.0        0.0  0.010000
4          0.0       0.0               0.0      0.0        0.0  0.020202

   Camera Store  Campanian Restaurant  Campground  Canal  Candy Store  \
0          0.01                   0.0         0.0    0.0          0.0
1          0.01                   0.0         0.0    0.0          0.0
2          0.00                   0.0         0.0    0.0          0.0
3          0.00                   0.0         0.0    0.0          0.0
4          0.00                   0.0         0.0    0.0          0.0
```

	Caribbean Restaurant	Castle	Cemetery	Cheese Shop	Chinese Restaurant
0	0.0	0.01	0.0	0.0	0.0
1	0.0	0.00	0.0	0.0	0.0
2	0.0	0.00	0.0	0.0	0.0
3	0.0	0.00	0.0	0.0	0.0
4	0.0	0.00	0.0	0.0	0.0

	Chocolate Shop	Church	City	Climbing Gym	Clothing Store	Cocktail Bar
0	0.01	0.00	0.0	0.0	0.01	0.010000
1	0.01	0.01	0.0	0.0	0.00	0.010000
2	0.00	0.02	0.0	0.0	0.02	0.010000
3	0.00	0.00	0.0	0.0	0.02	0.030000
4	0.00	0.00	0.0	0.0	0.00	0.010101

	Coffee Shop	College Arts Building	College Cafeteria	Comedy Club
0	0.020000	0.01	0.0	0.0
1	0.020000	0.00	0.0	0.0
2	0.030000	0.00	0.0	0.0
3	0.010000	0.01	0.0	0.0
4	0.020202	0.00	0.0	0.0

	Comic Shop	Convenience Store	Convention Center	Cosmetics Shop
0	0.0	0.0	0.0	0.010000
1	0.0	0.0	0.0	0.020000
2	0.0	0.0	0.0	0.010000
3	0.0	0.0	0.0	0.020000
4	0.0	0.0	0.0	0.020202

	Coworking Space	Creperie	Cultural Center	Cupcake Shop	Dance Studio
0	0.0	0.0	0.0	0.01	0.0
1	0.0	0.0	0.0	0.00	0.0
2	0.0	0.0	0.0	0.00	0.0
3	0.0	0.0	0.0	0.01	0.0
4	0.0	0.0	0.0	0.00	0.0

	Deli / Bodega	Department Store	Design Studio	Dessert Shop
0	0.0	0.000000	0.0	0.030000
1	0.0	0.010000	0.0	0.020000
2	0.0	0.010000	0.0	0.000000
3	0.0	0.010000	0.0	0.020000
4	0.0	0.010101	0.0	0.020202

	Dim Sum Restaurant	Diner	Discount Store	Dive Bar	Doner Restaurant
0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0

```
3                0.0    0.0             0.0       0.0              0.0
4                0.0    0.0             0.0       0.0              0.0

   Electronics Store  Empanada Restaurant  Event Space  Exhibit  Fabric Shop  \
0           0.000000                  0.0          0.0      0.0          0.0
1           0.010000                  0.0          0.0      0.0          0.0
2           0.010000                  0.0          0.0      0.0          0.0
3           0.010000                  0.0          0.0      0.0          0.0
4           0.010101                  0.0          0.0      0.0          0.0

   Falafel Restaurant  Farm  Farmers Market  Fast Food Restaurant  \
0                 0.0   0.0             0.0                   0.0
1                 0.0   0.0             0.0                   0.0
2                 0.0   0.0             0.0                   0.0
3                 0.0   0.0             0.0                   0.0
4                 0.0   0.0             0.0                   0.0

   Filipino Restaurant  Film Studio  Fish & Chips Shop  Fish Market  \
0                  0.0          0.0                0.0          0.0
1                  0.0          0.0                0.0          0.0
2                  0.0          0.0                0.0          0.0
3                  0.0          0.0                0.0          0.0
4                  0.0          0.0                0.0          0.0

   Flea Market  Flower Shop  Food  Food & Drink Shop  Food Court  Food Truck  \
0     0.000000          0.0   0.0               0.00         0.0         0.0
1     0.000000          0.0   0.0               0.01         0.0         0.0
2     0.000000          0.0   0.0               0.01         0.0         0.0
3     0.000000          0.0   0.0               0.01         0.0         0.0
4     0.010101          0.0   0.0               0.00         0.0         0.0

   Football Stadium  Fountain  French Restaurant  Fried Chicken Joint  \
0               0.0      0.01                0.0                  0.0
1               0.0      0.00                0.0                  0.0
2               0.0      0.00                0.0                  0.0
3               0.0      0.00                0.0                  0.0
4               0.0      0.00                0.0                  0.0

   Friterie  Frozen Yogurt Shop  Furniture / Home Store  Gaming Cafe  Garden  \
0       0.0                 0.0                0.000000          0.0     0.0
1       0.0                 0.0                0.000000          0.0     0.0
2       0.0                 0.0                0.010000          0.0     0.0
3       0.0                 0.0                0.000000          0.0     0.0
4       0.0                 0.0                0.010101          0.0     0.0

   Garden Center  Gas Station  Gastropub  Gay Bar  General Entertainment  \
0            0.0          0.0        0.0      0.0                    0.0
```

```
1            0.0          0.0        0.0      0.0                    0.0
2            0.0          0.0        0.0      0.0                    0.0
3            0.0          0.0        0.0      0.0                    0.0
4            0.0          0.0        0.0      0.0                    0.0

   German Restaurant  Gift Shop  Golf Course  Gourmet Shop  \
0                0.0   0.030000          0.0      0.010000
1                0.0   0.030000          0.0      0.020000
2                0.0   0.020000          0.0      0.020000
3                0.0   0.010000          0.0      0.000000
4                0.0   0.010101          0.0      0.010101

   Government Building  Greek Restaurant  Grocery Store  Gym  \
0                  0.0               0.0            0.0  0.0
1                  0.0               0.0            0.0  0.0
2                  0.0               0.0            0.0  0.0
3                  0.0               0.0            0.0  0.0
4                  0.0               0.0            0.0  0.0

   Gym / Fitness Center  Gym Pool  Harbor / Marina  Hardware Store  \
0                  0.00       0.0              0.0             0.0
1                  0.00       0.0              0.0             0.0
2                  0.01       0.0              0.0             0.0
3                  0.00       0.0              0.0             0.0
4                  0.00       0.0              0.0             0.0

   Hawaiian Restaurant  Health Food Store  Historic Site  History Museum  \
0                  0.0                0.0           0.00             0.0
1                  0.0                0.0           0.00             0.0
2                  0.0                0.0           0.01             0.0
3                  0.0                0.0           0.00             0.0
4                  0.0                0.0           0.00             0.0

   Hobby Shop  Hockey Arena    Hostel    Hotel  Hotel Bar  IT Services  \
0         0.0           0.0  0.000000  0.08000        0.0          0.0
1         0.0           0.0  0.000000  0.10000        0.0          0.0
2         0.0           0.0  0.010000  0.09000        0.0          0.0
3         0.0           0.0  0.000000  0.10000        0.0          0.0
4         0.0           0.0  0.010101  0.10101        0.0          0.0

   Ice Cream Shop  Indian Restaurant  Indie Movie Theater  Indie Theater  \
0        0.080000                0.0                  0.0            0.0
1        0.020000                0.0                  0.0            0.0
2        0.050000                0.0                  0.0            0.0
3        0.010000                0.0                  0.0            0.0
4        0.040404                0.0                  0.0            0.0
```

```
   Intersection  Irish Pub  Italian Restaurant  Japanese Restaurant  \
0           0.0        0.0            0.110000             0.010000
1           0.0        0.0            0.040000             0.010000
2           0.0        0.0            0.050000             0.010000
3           0.0        0.0            0.050000             0.020000
4           0.0        0.0            0.080808             0.020202

   Jazz Club  Jewelry Store  Juice Bar  Karaoke Bar  Kebab Restaurant  \
0        0.0       0.000000        0.0         0.01               0.0
1        0.0       0.010000        0.0         0.00               0.0
2        0.0       0.000000        0.0         0.00               0.0
3        0.0       0.010000        0.0         0.00               0.0
4        0.0       0.010101        0.0         0.00               0.0

   Kids Store  Kitchen Supply Store  Korean Restaurant  \
0         0.0              0.020000               0.00
1         0.0              0.010000               0.01
2         0.0              0.010000               0.01
3         0.0              0.000000               0.00
4         0.0              0.010101               0.00

   Latin American Restaurant  Lebanese Restaurant  Lombard Restaurant  \
0                        0.0                  0.0                 0.0
1                        0.0                  0.0                 0.0
2                        0.0                  0.0                 0.0
3                        0.0                  0.0                 0.0
4                        0.0                  0.0                 0.0

     Lounge  Market  Martial Arts School  Mattress Store  Medical Center  \
0  0.010000     0.0                  0.0             0.0             0.0
1  0.020000     0.0                  0.0             0.0             0.0
2  0.010000     0.0                  0.0             0.0             0.0
3  0.020000     0.0                  0.0             0.0             0.0
4  0.010101     0.0                  0.0             0.0             0.0

   Mediterranean Restaurant  Men's Store  Metro Station  Mexican Restaurant  \
0                      0.00     0.010000            0.0            0.000000
1                      0.01     0.020000            0.0            0.000000
2                      0.00     0.000000            0.0            0.000000
3                      0.01     0.030000            0.0            0.000000
4                      0.00     0.010101            0.0            0.010101

   Middle Eastern Restaurant  Miscellaneous Shop  Mobile Phone Shop  \
0                        0.0                 0.0                0.0
1                        0.0                 0.0                0.0
2                        0.0                 0.0                0.0
3                        0.0                 0.0                0.0
```

```
4                       0.0                0.0                0.0

   Mongolian Restaurant  Monument / Landmark  Moroccan Restaurant  \
0                   0.0             0.040000                  0.0
1                   0.0             0.040000                  0.0
2                   0.0             0.040000                  0.0
3                   0.0             0.030000                  0.0
4                   0.0             0.020202                  0.0

   Motorcycle Shop  Movie Theater  Multiplex    Museum  Music School  \
0              0.0       0.000000        0.0  0.010000      0.000000
1              0.0       0.000000        0.0  0.000000      0.000000
2              0.0       0.000000        0.0  0.000000      0.000000
3              0.0       0.000000        0.0  0.010000      0.000000
4              0.0       0.010101        0.0  0.010101      0.010101

   Music Store  Music Venue  Neighborhood  Nightclub  Noodle House  Office  \
0          0.0          0.0           0.0        0.0           0.0     0.0
1          0.0          0.0           0.0        0.0           0.0     0.0
2          0.0          0.0           0.0        0.0           0.0     0.0
3          0.0          0.0           0.0        0.0           0.0     0.0
4          0.0          0.0           0.0        0.0           0.0     0.0

   Opera House  Optical Shop  Other Nightlife  Outdoor Sculpture  \
0     0.010000           0.0              0.0                0.0
1     0.010000           0.0              0.0                0.0
2     0.010000           0.0              0.0                0.0
3     0.010000           0.0              0.0                0.0
4     0.010101           0.0              0.0                0.0

   Outdoors & Recreation  Outlet Mall  Outlet Store  Palace  \
0                    0.0          0.0           0.0     0.0
1                    0.0          0.0           0.0     0.0
2                    0.0          0.0           0.0     0.0
3                    0.0          0.0           0.0     0.0
4                    0.0          0.0           0.0     0.0

   Paper / Office Supplies Store      Park  Pastry Shop  Pedestrian Plaza  \
0                            0.0  0.010000     0.010000              0.00
1                            0.0  0.000000     0.010000              0.00
2                            0.0  0.030000     0.010000              0.00
3                            0.0  0.020000     0.010000              0.01
4                            0.0  0.030303     0.010101              0.00

   Performing Arts Venue  Perfume Shop  Persian Restaurant  Pet Store  \
0                    0.0          0.00                 0.0        0.0
1                    0.0          0.01                 0.0        0.0
```

```
2                   0.0          0.00                 0.0        0.0
3                   0.0          0.01                 0.0        0.0
4                   0.0          0.00                 0.0        0.0

   Pharmacy  Photography Lab  Piadineria  Pizza Place  Planetarium  Platform  \
0       0.0              0.0         0.0     0.030000          0.0       0.0
1       0.0              0.0         0.0     0.010000          0.0       0.0
2       0.0              0.0         0.0     0.030000          0.0       0.0
3       0.0              0.0         0.0     0.010000          0.0       0.0
4       0.0              0.0         0.0     0.050505          0.0       0.0

   Playground     Plaza  Pool  Pool Hall  Print Shop  Pub  Public Art  \
0         0.0  0.080000   0.0        0.0         0.0  0.0         0.0
1         0.0  0.090000   0.0        0.0         0.0  0.0         0.0
2         0.0  0.090000   0.0        0.0         0.0  0.0         0.0
3         0.0  0.070000   0.0        0.0         0.0  0.0         0.0
4         0.0  0.070707   0.0        0.0         0.0  0.0         0.0

   Puglia Restaurant  Racecourse  Racetrack  Radio Station  Ramen Restaurant  \
0                0.0         0.0        0.0            0.0               0.0
1                0.0         0.0        0.0            0.0               0.0
2                0.0         0.0        0.0            0.0               0.0
3                0.0         0.0        0.0            0.0               0.0
4                0.0         0.0        0.0            0.0               0.0

   Record Shop  Recording Studio  Recreation Center  \
0          0.0               0.0                0.0
1          0.0               0.0                0.0
2          0.0               0.0                0.0
3          0.0               0.0                0.0
4          0.0               0.0                0.0

   Residential Building (Apartment / Condo)  Rest Area  Restaurant  River  \
0                                       0.0        0.0    0.010000    0.0
1                                       0.0        0.0    0.010000    0.0
2                                       0.0        0.0    0.020000    0.0
3                                       0.0        0.0    0.010000    0.0
4                                       0.0        0.0    0.010101    0.0

   Road  Rock Club  Roman Restaurant  Russian Restaurant  Salad Place  \
0  0.00        0.0               0.0                 0.0         0.00
1  0.01        0.0               0.0                 0.0         0.00
2  0.00        0.0               0.0                 0.0         0.01
3  0.01        0.0               0.0                 0.0         0.00
4  0.00        0.0               0.0                 0.0         0.00

   Salon / Barbershop  Sandwich Place  Sardinian Restaurant  \
```

```
0                0.01            0.02                  0.00
1                0.01            0.00                  0.00
2                0.01            0.01                  0.01
3                0.01            0.00                  0.00
4                0.00            0.00                  0.00

   Sauna / Steam Room  Scenic Lookout  Science Museum  Seafood Restaurant  \
0                 0.0        0.000000             0.0            0.000000
1                 0.0        0.010000             0.0            0.000000
2                 0.0        0.010000             0.0            0.000000
3                 0.0        0.010000             0.0            0.000000
4                 0.0        0.010101             0.0            0.030303

   Shipping Store  Shoe Store  Shopping Mall  Shopping Plaza  \
0             0.0        0.01            0.0             0.0
1             0.0        0.00            0.0             0.0
2             0.0        0.00            0.0             0.0
3             0.0        0.00            0.0             0.0
4             0.0        0.00            0.0             0.0

   Sicilian Restaurant  Skate Park  Smoke Shop  Snack Place  Soccer Field  \
0                  0.0         0.0        0.01          0.0           0.0
1                  0.0         0.0        0.00          0.0           0.0
2                  0.0         0.0        0.00          0.0           0.0
3                  0.0         0.0        0.00          0.0           0.0
4                  0.0         0.0        0.00          0.0           0.0

   Soccer Stadium  Social Club  South American Restaurant  \
0             0.0          0.0                        0.0
1             0.0          0.0                        0.0
2             0.0          0.0                        0.0
3             0.0          0.0                        0.0
4             0.0          0.0                        0.0

   South Tyrolean Restaurant   Spa  Spanish Restaurant  Speakeasy  \
0                        0.0  0.01            0.000000        0.0
1                        0.0  0.00            0.000000        0.0
2                        0.0  0.00            0.000000        0.0
3                        0.0  0.00            0.000000        0.0
4                        0.0  0.00            0.010101        0.0

   Sporting Goods Shop  Sports Bar  Sports Club  Sri Lankan Restaurant  \
0             0.000000    0.000000          0.0                    0.0
1             0.020000    0.000000          0.0                    0.0
2             0.020000    0.000000          0.0                    0.0
3             0.000000    0.000000          0.0                    0.0
4             0.010101    0.010101          0.0                    0.0
```

	Stadium	Stationery Store	Steakhouse	Street Art	Strip Club	Supermarket	\
0	0.0	0.01	0.0	0.0	0.0	0.000000	
1	0.0	0.00	0.0	0.0	0.0	0.000000	
2	0.0	0.00	0.0	0.0	0.0	0.000000	
3	0.0	0.00	0.0	0.0	0.0	0.000000	
4	0.0	0.00	0.0	0.0	0.0	0.010101	

	Sushi Restaurant	Szechuan Restaurant	Tapas Restaurant	Tea Room	\
0	0.020000	0.0	0.01	0.01	
1	0.000000	0.0	0.00	0.00	
2	0.000000	0.0	0.00	0.00	
3	0.000000	0.0	0.00	0.00	
4	0.010101	0.0	0.00	0.00	

	Tennis Court	Tennis Stadium	Thai Restaurant	Theater	\
0	0.0	0.0	0.0	0.03	
1	0.0	0.0	0.0	0.00	
2	0.0	0.0	0.0	0.01	
3	0.0	0.0	0.0	0.00	
4	0.0	0.0	0.0	0.00	

	Theme Park Ride / Attraction	Thrift / Vintage Store	Toy / Game Store	\
0	0.0	0.00	0.000000	
1	0.0	0.00	0.010000	
2	0.0	0.01	0.000000	
3	0.0	0.00	0.010000	
4	0.0	0.00	0.010101	

	Track	Trail	Train	Train Station	Tram Station	Trattoria/Osteria	\
0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	0.0	0.0	

	Tunnel	Turkish Restaurant	Tuscan Restaurant	\
0	0.0	0.0	0.0	
1	0.0	0.0	0.0	
2	0.0	0.0	0.0	
3	0.0	0.0	0.0	
4	0.0	0.0	0.0	

	Vegetarian / Vegan Restaurant	Video Game Store	Video Store	\
0	0.0	0.0	0.0	
1	0.0	0.0	0.0	
2	0.0	0.0	0.0	

```
3                           0.0                0.0          0.0
4                           0.0                0.0          0.0

   Vietnamese Restaurant  Volleyball Court  Warehouse Store  Water Park  \
0                    0.0               0.0              0.0         0.0
1                    0.0               0.0              0.0         0.0
2                    0.0               0.0              0.0         0.0
3                    0.0               0.0              0.0         0.0
4                    0.0               0.0              0.0         0.0

   Wine Bar  Wine Shop  Winery  Women's Store  Yoga Studio
0  0.020000        0.0     0.0           0.00          0.0
1  0.020000        0.0     0.0           0.01          0.0
2  0.010000        0.0     0.0           0.00          0.0
3  0.010000        0.0     0.0           0.04          0.0
4  0.010101        0.0     0.0           0.00          0.0
```

2.15.4 Top 10 most common venues for each neighborhood

[67]:
```
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
areaColumns = ['Borough code', 'Neighborhoods', 'Latitude',␣
 ↪'Longitude','Rent_price','Buy_price']
freqColumns = []
for ind in np.arange(num_top_venues):
    try:
        freqColumns.append('{}{} Most Common Venue'.format(ind+1,␣
 ↪indicators[ind]))
    except:
        freqColumns.append('{}th Most Common Venue'.format(ind+1))
columns = areaColumns+freqColumns

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Borough code'] = Milan_grouped['Borough code']
neighborhoods_venues_sorted['Neighborhoods'] = Milan_grouped['Neighborhoods']
neighborhoods_venues_sorted['Latitude'] = Milan_grouped['Latitude']
neighborhoods_venues_sorted['Longitude'] = Milan_grouped['Longitude']
neighborhoods_venues_sorted['Rent_price'] = Milan_grouped['Rent_price']
neighborhoods_venues_sorted['Buy_price'] = Milan_grouped['Buy_price']


for ind in np.arange(Milan_grouped.shape[0]):
```

```
    row_categories = Milan_grouped.iloc[ind, :].iloc[6:]
    row_categories_sorted = row_categories.sort_values(ascending=False)
    neighborhoods_venues_sorted.iloc[ind, 6:] = row_categories_sorted.index.
↪values[0:num_top_venues]

print(neighborhoods_venues_sorted.shape)
neighborhoods_venues_sorted.head()
```

```
(98, 16)
```

[67]:

	Borough code	Neighborhoods	Latitude	Longitude	Rent_price	Buy_price	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	B12	CAIROLI	45.468701	9.181697	33.333333	10533.333333	Italian Restaurant	Hotel	Ice Cream Shop	Plaza	Monument / Landmark	Café	Art Museum	Dessert Shop	Gift Shop	Pizza Place
1	B12	DUOMO	45.464585	9.189670	33.333333	10533.333333	Boutique	Hotel	Plaza	Italian Restaurant	Monument / Landmark	Art Museum	Bakery	Gift Shop	Coffee Shop	Bar
2	B12	MISSORI	45.459828	9.189555	33.333333	10533.333333	Hotel	Plaza	Ice Cream Shop	Café	Italian Restaurant	Boutique	Monument / Landmark	Park	Coffee Shop	Pizza Place
3	B12	MONTENAPOLEONE	45.470015	9.192868	33.333333	10533.333333	Boutique	Hotel	Plaza	Italian Restaurant	Women's Store	Bakery	Men's Store	Monument / Landmark	Cocktail Bar	Lounge
4	B12	SAN BABILA	45.466521	9.197529	33.333333	10533.333333	Hotel	Italian Restaurant	Plaza	Pizza Place	Boutique	Ice Cream Shop	Bakery	Park	Seafood Restaurant	Art Gallery

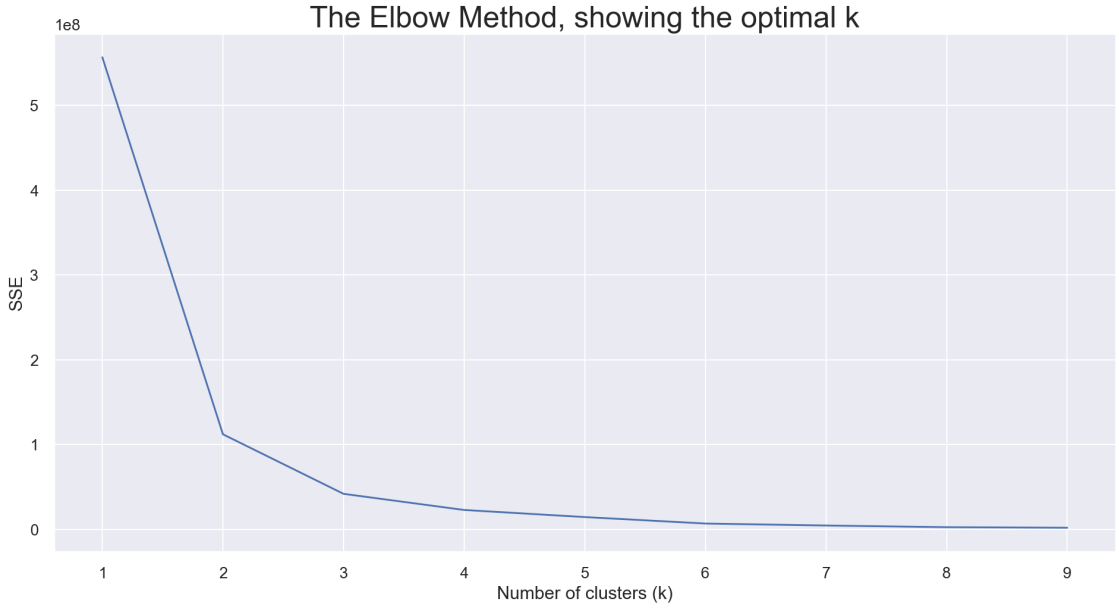
2.15.5 Clustering Neighborhoods

Finding the optimal number of clusters

In order to try to determine the optimal number of clusters (k), two different methods are going to be used – the elbow method and the silhouette score. Both of the analyses will be run on a range between 2 and 9 clusters.

Let's start with the elbow method:

[68]:
```
Milan_grouped_clustering = Milan_grouped.drop(["Borough code", "Neighborhoods",␣
 ↪"Latitude", "Longitude"], 1)
sse = {}
for k in range(1, 10):
    kmeans = KMeans(n_clusters=k, max_iter=1000, random_state=1).
 ↪fit(Milan_grouped_clustering)
    sse[k] = kmeans.inertia_ # Inertia: Sum of distances of samples to their␣
 ↪closest cluster center
plt.figure(figsize=(16,8))
plt.plot(list(sse.keys()), list(sse.values()), 'bx-')
plt.xlabel("Number of clusters (k)", fontsize= 15)
plt.ylabel("SSE", fontsize= 15)
plt.title('The Elbow Method, showing the optimal k', fontsize= 25)
plt.grid(True)
plt.show()
```



A bend in the curve at k = 2 it's an indication that 2 the best number of clusters.

Let's try the silhouette method:

[69]:
```
for n_clusters in range(2, 10):
    clusterer = KMeans(n_clusters=n_clusters)
    preds = clusterer.fit_predict(Milan_grouped_clustering)
    centers = clusterer.cluster_centers_

    score = silhouette_score(Milan_grouped_clustering, preds)
    print("For n_clusters = {}, silhouette score is {})".format(n_clusters,␣
 ↪score))
```

```
For n_clusters = 2, silhouette score is 0.7635447010527375)
For n_clusters = 3, silhouette score is 0.7556555420460048)
For n_clusters = 4, silhouette score is 0.7331693135077795)
For n_clusters = 5, silhouette score is 0.7252673998054638)
For n_clusters = 6, silhouette score is 0.7579938931228684)
For n_clusters = 7, silhouette score is 0.7012728825715278)
For n_clusters = 8, silhouette score is 0.7036627916355187)
For n_clusters = 9, silhouette score is 0.7099134882542403)
```

The silhouette method confirm that the optimal number of clusters for the dataset is k = 2, since it has the highest score.

[70]:
```
# set number of clusters
kclusters = 2

Milan_grouped_clustering = Milan_grouped.drop(["Borough code", "Neighborhoods",␣
 ↪"Latitude", "Longitude"], 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).
 ↪fit(Milan_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_
```

[70]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0,
       1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int32)
```

2.15.6 Add cluster_lablel column to Milan_merged dataframe

[71]:
```
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
neighborhoods_venues_sorted.sort_values(by=['Cluster Labels'], inplace=True)
print(neighborhoods_venues_sorted.shape)
neighborhoods_venues_sorted.head()
```

(98, 17)

[71]:

	Cluster Labels	Borough code	Neighborhoods	Latitude	Longitude	Rent_price	Buy_price	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
48	0	D15	VIA CARLO MAROCHETTI	45.439354	9.225242	8.350	2625.0	Pizza Place	Nightclub	Plaza	Café	Italian Restaurant	Dessert Shop	Hotel	Japanese Restaurant	Soccer Field	Theater
70	0	D31	BOVISASCA	45.515842	9.153778	8.400	2775.0	Supermarket	Pizza Place	Gym / Fitness Center	Café	Italian Restaurant	Cosmetics Shop	Clothing Store	Music Store	Soccer Field	Shopping Plaza
69	0	D31	BAUSAN	45.502634	9.166374	8.400	2775.0	Café	Italian Restaurant	Pizza Place	Ice Cream Shop	Plaza	Diner	Supermarket	Bar	Piadineria	Hotel
68	0	D30	EXPO	45.522917	9.094042	8.875	2550.0	Sporting Goods Shop	Italian Restaurant	Café	Sandwich Place	Food Court	Diner	Train Station	Fast Food Restaurant	Platform	Pedestrian Plaza
67	0	D30	CERTOSA	45.502489	9.127848	8.875	2550.0	Supermarket	Café	Tram Station	Italian Restaurant	Pizza Place	Gym	Bistro	Pet Store	Bakery	Seafood Restaurant

2.15.7 Visualize the resulting clusters

[72]:
```
map_clusters = folium.Map(location=[lat, lon], zoom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]
```

```
def color(cluster):
    if cluster == 0:
        col_= 'green'
    else:
        col_='orange'
    return col_

# add markers to the map
markers_colors = []
for lat, lon, b_code, neigh, cluster in␣
 ↪zip(neighborhoods_venues_sorted['Latitude'],␣
 ↪neighborhoods_venues_sorted['Longitude'],␣
 ↪neighborhoods_venues_sorted['Borough code'],␣
 ↪neighborhoods_venues_sorted['Neighborhoods'],␣
 ↪neighborhoods_venues_sorted['Cluster Labels']):
    label = folium.Popup('{} (Borough code: {}) - Cluster {}'.format(neigh,␣
 ↪b_code, cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=3,
        popup=label,
        color= color(cluster),
        parse_html=False, legend_name='SCALE',
        fill=True,
        fill_opacity=0.8).add_to(map_clusters)

legend_html = '''
    <div style="position: fixed;
        bottom: 10px; left: 10px; width: 100px; height: 70px;
        border:2px solid grey; z-index:9999; font-size:11px;">  Legend <br>
          <b> Cluster 1 </b>   <i class="fa fa-map-marker fa-2x"␣
 ↪style="color:green"></i><br>
          <b> Cluster 2 </b>   <i class="fa fa-map-marker fa-2x"␣
 ↪style="color:orange"></i><br>
    </div>
    '''
map_clusters.get_root().html.add_child(folium.Element(legend_html))

map_clusters
```

[72]: <folium.folium.Map at 0x1a1a618400>

2.15.8 Examine Clusters

cluster 1:

[73]:
```
neighborhoods_venues_sorted.loc[neighborhoods_venues_sorted['Cluster Labels']␣
↪== 0, neighborhoods_venues_sorted.columns[[1] + list(range(5,␣
↪neighborhoods_venues_sorted.shape[1]))]]
```

[73]:

	Borough	code	Rent_price	Buy_price	1st Most Common Venue
48		D15	8.350000	2625.0	Pizza Place
70		D31	8.400000	2775.0	Supermarket
69		D31	8.400000	2775.0	Café
68		D30	8.875000	2550.0	Sporting Goods Shop
67		D30	8.875000	2550.0	Supermarket
66		D30	8.875000	2550.0	Bus Station
65		D30	8.875000	2550.0	Café
64		D28	11.050000	3350.0	Park
63		D28	11.050000	3350.0	Italian Restaurant
62		D28	11.050000	3350.0	Café
71		D31	8.400000	2775.0	Café
61		D25	9.175000	2475.0	Italian Restaurant
59		D25	9.175000	2475.0	Pizza Place
58		D25	9.175000	2475.0	Pizza Place
57		D21	10.400000	2712.5	Park
56		D21	10.400000	2712.5	Italian Restaurant
55		D21	10.400000	2712.5	Italian Restaurant
54		D20	10.500000	3012.5	Pizza Place
53		D20	10.500000	3012.5	Italian Restaurant
52		D20	10.500000	3012.5	Soccer Field
51		D18	7.925000	2762.5	Pizza Place
60		D25	9.175000	2475.0	Pizza Place
72		D32	8.175000	2575.0	Café
73		D32	8.175000	2575.0	Bus Station
74		D32	8.175000	2575.0	Pizza Place
95		E7	7.250000	2212.5	Clothing Store
94		E7	7.250000	2212.5	Tram Station
93		E6	8.200000	2262.5	Park
92		E6	8.200000	2262.5	Gym / Fitness Center
91		E6	8.200000	2262.5	Soccer Stadium
90		E6	8.200000	2262.5	Pizza Place
89		E5	7.450000	2225.0	Supermarket
88		E5	7.450000	2225.0	Neighborhood
87		E5	7.450000	2225.0	Pizza Place
86		D36	9.425000	2812.5	Italian Restaurant
85		D36	9.425000	2812.5	Italian Restaurant
84		D36	9.425000	2812.5	Pizza Place
83		D35	9.050000	2637.5	Italian Restaurant
82		D35	9.050000	2637.5	Italian Restaurant
81		D35	9.050000	2637.5	Italian Restaurant
80		D35	9.050000	2637.5	Café
79		D34	9.125000	2862.5	Italian Restaurant

78	D34	9.125000	2862.5	Café
77	D33	8.875000	2725.0	Pizza Place
76	D33	8.875000	2725.0	Pizza Place
75	D33	8.875000	2725.0	Steakhouse
50	D18	7.925000	2762.5	Café
49	D18	7.925000	2762.5	Italian Restaurant
97	E8	6.650000	2062.5	Hotel
47	D15	8.350000	2625.0	Italian Restaurant
17	B20	19.075000	5475.0	Italian Restaurant
18	B21	15.050000	5600.0	Italian Restaurant
19	B21	15.050000	5600.0	Italian Restaurant
20	C12	18.725000	5400.0	Italian Restaurant
21	C12	18.725000	5400.0	Italian Restaurant
96	E8	6.650000	2062.5	Metro Station
24	C15	10.375000	3575.0	Italian Restaurant
25	C16	11.700000	4025.0	Italian Restaurant
26	C16	11.700000	4025.0	Italian Restaurant
27	C16	11.700000	4025.0	Italian Restaurant
28	C17	16.983333	5350.0	Italian Restaurant
30	C17	16.983333	5350.0	Italian Restaurant
31	C18	12.275000	4250.0	Italian Restaurant
32	C18	12.275000	4250.0	Italian Restaurant
29	C17	16.983333	5350.0	Italian Restaurant
34	C19	12.675000	4325.0	Café
46	D15	8.350000	2625.0	Platform
45	D15	8.350000	2625.0	Tram Station
44	D13	7.625000	2600.0	Italian Restaurant
43	D13	7.625000	2600.0	Pizza Place
33	C18	12.275000	4250.0	Italian Restaurant
41	D12	10.600000	3450.0	Pizza Place
42	D13	7.625000	2600.0	Italian Restaurant
39	D12	10.600000	3450.0	Pizza Place
38	D10	9.350000	2900.0	Pizza Place
37	D10	9.350000	2900.0	Chinese Restaurant
36	D10	9.350000	2900.0	Pizza Place
35	C19	12.675000	4325.0	Cocktail Bar
40	D12	10.600000	3450.0	Pizza Place

	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	\
48	Nightclub	Plaza	Café	
70	Pizza Place	Gym / Fitness Center	Café	
69	Italian Restaurant	Pizza Place	Ice Cream Shop	
68	Italian Restaurant	Café	Sandwich Place	
67	Café	Tram Station	Italian Restaurant	
66	Café	Metro Station	Park	
65	Italian Restaurant	Supermarket	Pizza Place	
64	Supermarket	Pizza Place	Hotel	

63	Tram Station	Supermarket	Hotel
62	Italian Restaurant	Plaza	Soccer Stadium
71	Pizza Place	Italian Restaurant	Supermarket
61	Ice Cream Shop	Pizza Place	Dessert Shop
59	Café	Italian Restaurant	Supermarket
58	Supermarket	Metro Station	Gym
57	Supermarket	Bus Stop	Soccer Field
56	Café	Supermarket	Pizza Place
55	Park	Gym	Café
54	Italian Restaurant	Café	Tram Station
53	Pizza Place	Cocktail Bar	Japanese Restaurant
52	Café	Japanese Restaurant	Arts & Crafts Store
51	Café	Tram Station	Supermarket
60	Café	Soccer Stadium	Seafood Restaurant
72	Italian Restaurant	Pizza Place	Hotel
73	Italian Restaurant	Gym / Fitness Center	Pizza Place
74	Pub	Tram Station	Hotel
95	Park	Pizza Place	Shoe Store
94	Soccer Field	Japanese Restaurant	Park
93	Pizza Place	Bakery	Pub
92	Pub	Pizza Place	Plaza
91	Park	Italian Restaurant	Athletics & Sports
90	Fast Food Restaurant	Asian Restaurant	Soccer Field
89	Soccer Field	Bus Station	Water Park
88	Hotel	Soccer Field	Snack Place
87	Ice Cream Shop	Fast Food Restaurant	Soccer Field
86	Pizza Place	Ice Cream Shop	Chinese Restaurant
85	Ice Cream Shop	Japanese Restaurant	Chinese Restaurant
84	Park	Café	Supermarket
83	Supermarket	Café	Park
82	Hotel	Café	Seafood Restaurant
81	Pizza Place	Café	Hotel
80	Pizza Place	Park	Italian Restaurant
79	Pizza Place	Café	Tram Station
78	Pizza Place	Italian Restaurant	Pub
77	Tram Station	Restaurant	Cocktail Bar
76	Café	Italian Restaurant	Hotel
75	Pizza Place	Hotel	Italian Restaurant
50	Pizza Place	Tram Station	Hotel
49	Tram Station	Supermarket	Café
97	Café	Japanese Restaurant	Plaza
47	Park	Tram Station	Theater
17	Cocktail Bar	Ice Cream Shop	Seafood Restaurant
18	Pizza Place	Hotel	Sandwich Place
19	Cocktail Bar	Ice Cream Shop	Café
20	Pizza Place	Plaza	Hotel
21	Pizza Place	Dessert Shop	Ice Cream Shop

96	Restaurant	Brewery	Rock Club	
24	Chinese Restaurant	Pizza Place	Restaurant	
25	Café	Pizza Place	Japanese Restaurant	
26	Pizza Place	Chinese Restaurant	Seafood Restaurant	
27	Cocktail Bar	Chinese Restaurant	Pizza Place	
28	Plaza	Ice Cream Shop	Pizza Place	
30	Ice Cream Shop	Pizza Place	Sandwich Place	
31	Cocktail Bar	Ice Cream Shop	Café	
32	Café	Cocktail Bar	Plaza	
29	Ice Cream Shop	Seafood Restaurant	Pizza Place	
34	Italian Restaurant	Pizza Place	Pub	
46	Pizza Place	Italian Restaurant	Café	
45	Shipping Store	Hotel	Pizza Place	
44	Nightclub	Café	Steakhouse	
43	Café	Italian Restaurant	Bakery	
33	Cocktail Bar	Ice Cream Shop	Seafood Restaurant	
41	Café	Italian Restaurant	Supermarket	
42	Pizza Place	Café	Seafood Restaurant	
39	Café	Italian Restaurant	Seafood Restaurant	
38	Italian Restaurant	Café	Supermarket	
37	Italian Restaurant	Café	Dessert Shop	
36	Italian Restaurant	Café	Restaurant	
35	Wine Bar	Pizza Place	Italian Restaurant	
40	Italian Restaurant	Café	Seafood Restaurant	

	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	\
48	Italian Restaurant	Dessert Shop	Hotel	
70	Italian Restaurant	Cosmetics Shop	Clothing Store	
69	Plaza	Diner	Supermarket	
68	Food Court	Diner	Train Station	
67	Pizza Place	Gym	Bistro	
66	Noodle House	Asian Restaurant	Shoe Store	
65	Japanese Restaurant	Piadineria	Ice Cream Shop	
64	Tram Station	Bistro	Steakhouse	
63	Restaurant	Café	Park	
62	Metro Station	Supermarket	Pizza Place	
71	Plaza	Tram Station	Hotel	
61	Bar	Bakery	Cocktail Bar	
59	Restaurant	Arcade	Bar	
58	Bar	Bus Station	Bakery	
57	Café	Theater	Japanese Restaurant	
56	Restaurant	Cafeteria	Trail	
55	Supermarket	Hotel	Pizza Place	
54	Restaurant	Cocktail Bar	Supermarket	
53	Supermarket	Café	Restaurant	
52	Park	Tennis Stadium	Theater	
51	Hotel	Nightclub	Gym / Fitness Center	

60	Supermarket	Italian Restaurant	Sporting Goods Shop
72	Gym / Fitness Center	Supermarket	Soccer Field
73	Bar	Fast Food Restaurant	Supermarket
74	Bus Station	Gym	Supermarket
95	Bus Stop	Supermarket	Café
94	Lombard Restaurant	Supermarket	Restaurant
93	Sandwich Place	Athletics & Sports	Soccer Field
92	Grocery Store	Flower Shop	Dessert Shop
91	Café	Stadium	Bus Station
90	Café	Metro Station	Supermarket
89	Campground	Hotel	Filipino Restaurant
88	Gas Station	Supermarket	Restaurant
87	Café	Asian Restaurant	Clothing Store
86	Japanese Restaurant	Cocktail Bar	Hotel
85	Hotel	Dessert Shop	Trattoria/Osteria
84	Japanese Restaurant	Bar	Plaza
83	Bakery	Pizza Place	Toy / Game Store
82	Ice Cream Shop	Restaurant	Pizza Place
81	Bakery	Gym	Brewery
80	Soccer Field	Ice Cream Shop	Outlet Mall
79	Theater	Bakery	Brewery
78	Steakhouse	Supermarket	Tram Station
77	Ice Cream Shop	Plaza	Pub
76	Soccer Field	Sushi Restaurant	Supermarket
75	Sandwich Place	Multiplex	Bus Stop
50	Supermarket	Italian Restaurant	Nightclub
49	Ice Cream Shop	Pharmacy	Soccer Field
97	Steakhouse	Supermarket	Tennis Court
47	Hotel	Gym	Pizza Place
17	Japanese Restaurant	Hotel	Plaza
18	Ice Cream Shop	Café	Cocktail Bar
19	Seafood Restaurant	Plaza	Bistro
20	Ice Cream Shop	Cocktail Bar	Sandwich Place
21	Sandwich Place	Hotel	Restaurant
96	Food	Tennis Court	Chinese Restaurant
24	Café	Japanese Restaurant	Trattoria/Osteria
25	Hotel	Tram Station	Supermarket
26	Trattoria/Osteria	Japanese Restaurant	Restaurant
27	Japanese Restaurant	Seafood Restaurant	Hotel
28	Café	Sandwich Place	Bistro
30	Supermarket	Asian Restaurant	Bistro
31	Seafood Restaurant	Plaza	Bistro
32	Pizza Place	Ice Cream Shop	Bistro
29	Café	Japanese Restaurant	Hotel
34	Asian Restaurant	Park	Outlet Mall
46	Supermarket	Bus Stop	Pedestrian Plaza
45	Music Venue	Tennis Court	Park

44	Supermarket	Rock Club	Pizza Place	
43	Cocktail Bar	Dessert Shop	Supermarket	
33	Japanese Restaurant	Vegetarian / Vegan Restaurant	Hotel	
41	Bakery	Ice Cream Shop	Restaurant	
42	Dessert Shop	Hotel	Restaurant	
39	Hotel	Cocktail Bar	Bakery	
38	Dessert Shop	Bakery	Asian Restaurant	
37	Cocktail Bar	Platform	Bakery	
36	Tram Station	Bakery	Asian Restaurant	
35	Pub	Japanese Restaurant	Restaurant	
40	Tram Station	Hotel	Cocktail Bar	

	8th Most Common Venue	9th Most Common Venue	\
48	Japanese Restaurant	Soccer Field	
70	Music Store	Soccer Field	
69	Bar	Piadineria	
68	Fast Food Restaurant	Platform	
67	Pet Store	Bakery	
66	Sushi Restaurant	Pet Store	
65	Park	Smoke Shop	
64	Gym	Clothing Store	
63	Japanese Restaurant	Bookstore	
62	Bar	Cocktail Bar	
71	Nightclub	Ice Cream Shop	
61	Café	Japanese Restaurant	
59	Bakery	Hockey Arena	
58	Park	Athletics & Sports	
57	Restaurant	Tennis Stadium	
56	Electronics Store	Train Station	
55	Restaurant	Russian Restaurant	
54	Pub	Piadineria	
53	Pub	Ice Cream Shop	
52	Athletics & Sports	Trattoria/Osteria	
51	Gastropub	Restaurant	
60	Bakery	Metro Station	
72	Park	Pool Hall	
73	General Entertainment	Metro Station	
74	Bar	Theater	
95	Cosmetics Shop	Shopping Mall	
94	Café	Food Court	
93	Plaza	Ice Cream Shop	
92	Burger Joint	Food Court	
91	Nightclub	Metro Station	
90	Sandwich Place	City	
89	Film Studio	Fish & Chips Shop	
88	Mediterranean Restaurant	Juice Bar	
87	Trattoria/Osteria	Sandwich Place	

86	Café	Trattoria/Osteria	
85	Café	Restaurant	
84	Convenience Store	Bus Station	
83	Clothing Store	Trattoria/Osteria	
82	Bistro	Plaza	
81	Ice Cream Shop	Chinese Restaurant	
80	Supermarket	Clothing Store	
79	Restaurant	Plaza	
78	Japanese Restaurant	Plaza	
77	Bed & Breakfast	Café	
76	Pub	Salon / Barbershop	
75	Café	Shopping Mall	
50	Seafood Restaurant	Gym	
49	Breakfast Spot	Bowling Alley	
97	Park	Tram Station	
47	Soccer Field	Noodle House	
17	Restaurant	Café	
18	Vegetarian / Vegan Restaurant	Dessert Shop	
19	Bar	Japanese Restaurant	
20	Café	Coffee Shop	
21	Bookstore	Sushi Restaurant	
96	Pizza Place	Arts & Entertainment	
24	Hotel	Seafood Restaurant	
25	Seafood Restaurant	Restaurant	
26	Bistro	Bakery	
27	Sushi Restaurant	Trattoria/Osteria	
28	Clothing Store	Coffee Shop	
30	Café	Chinese Restaurant	
31	Bar	Japanese Restaurant	
32	Japanese Restaurant	Hotel	
29	Boutique	Art Gallery	
34	Fast Food Restaurant	Performing Arts Venue	
46	Restaurant	Spa	
45	Spa	Garden	
44	Buffet	Furniture / Home Store	
43	Bar	Ice Cream Shop	
33	Café	Boutique	
41	Bar	Asian Restaurant	
42	Sushi Restaurant	Plaza	
39	Tram Station	Ice Cream Shop	
38	Chinese Restaurant	Restaurant	
37	Japanese Restaurant	Athletics & Sports	
36	Nightclub	Supermarket	
35	Bar	Café	
40	Bakery	Sushi Restaurant	

	10th Most Common Venue

48	Theater
70	Shopping Plaza
69	Hotel
68	Pedestrian Plaza
67	Seafood Restaurant
66	Auto Garage
65	Bar
64	Seafood Restaurant
63	Shopping Mall
62	Coffee Shop
71	Gym / Fitness Center
61	Restaurant
59	Japanese Restaurant
58	Bus Stop
57	Bed & Breakfast
56	Tram Station
55	Cocktail Bar
54	Sushi Restaurant
53	Diner
52	Bakery
51	Chinese Restaurant
60	Restaurant
72	Hostel
73	Bakery
74	Beer Bar
95	Sports Club
94	Bus Stop
93	Adult Education Center
92	Food & Drink Shop
91	Tennis Court
90	Gym
89	Fish Market
88	Airport
87	Supermarket
86	Asian Restaurant
85	Cocktail Bar
84	Italian Restaurant
83	Electronics Store
82	Sushi Restaurant
81	Sandwich Place
80	Trattoria/Osteria
79	Breakfast Spot
78	Restaurant
77	Gym
76	Park
75	Tram Station
50	Restaurant

```
49             Chinese Restaurant
97                     Restaurant
47                          Café
17  Vegetarian / Vegan Restaurant
18                    Art Gallery
19                          Hotel
20                      Beer Bar
21             Chinese Restaurant
96                            Gym
24                         Bistro
25                          Plaza
26                           Café
27                         Bakery
28               Asian Restaurant
30                 Clothing Store
31                          Hotel
32               Ramen Restaurant
29                   Burger Joint
34                          Trail
46                      Cafeteria
45           Fast Food Restaurant
44                      Beer Bar
43                          Plaza
33               Sushi Restaurant
41                          Plaza
42                 Ice Cream Shop
39                          Plaza
38                            Bar
37                   Pizza Place
36                          Hotel
35  Vegetarian / Vegan Restaurant
40                    Music Venue
```

Cluster 2:

[74]:
```
neighborhoods_venues_sorted.loc[neighborhoods_venues_sorted['Cluster Labels']␣
↪== 1, neighborhoods_venues_sorted.columns[[1] + list(range(5,␣
↪neighborhoods_venues_sorted.shape[1]))]]
```

[74]:

	Borough code	Rent_price	Buy_price	1st Most Common Venue
1	B12	33.333333	10533.333333	Boutique
7	B16	24.433333	8566.666667	Italian Restaurant
2	B12	33.333333	10533.333333	Hotel
3	B12	33.333333	10533.333333	Boutique
4	B12	33.333333	10533.333333	Hotel
5	B13	21.750000	7333.333333	Pizza Place
6	B15	26.566667	9150.000000	Italian Restaurant

```
8           B16   24.433333   8566.666667    Italian Restaurant
23          C14   29.733333   9033.333333    Italian Restaurant
10          B17   20.866667   7100.000000    Italian Restaurant
11          B17   20.866667   7100.000000    Italian Restaurant
12          B18   20.875000   6650.000000    Italian Restaurant
13          B18   20.875000   6650.000000    Italian Restaurant
14          B18   20.875000   6650.000000              Boutique
15          B19   21.950000   7216.666667    Italian Restaurant
16          B19   21.950000   7216.666667    Italian Restaurant
22          C13   30.925000   9275.000000    Italian Restaurant
9           B16   24.433333   8566.666667                Plaza
0           B12   33.333333  10533.333333    Italian Restaurant

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
1                  Hotel                Plaza    Italian Restaurant
7          Ice Cream Shop                Plaza                 Hotel
2                  Plaza        Ice Cream Shop                 Café
3                  Hotel                Plaza    Italian Restaurant
4     Italian Restaurant                Plaza           Pizza Place
5          Ice Cream Shop         Dessert Shop    Italian Restaurant
6                  Hotel             Boutique        Ice Cream Shop
8          Ice Cream Shop                Café         Historic Site
23                 Hotel                Café    Chinese Restaurant
10           Cocktail Bar   Japanese Restaurant        Ice Cream Shop
11         Ice Cream Shop                Plaza               Theater
12         Ice Cream Shop                Hotel   Japanese Restaurant
13                 Hotel        Ice Cream Shop           Pizza Place
14                 Hotel    Italian Restaurant          Cocktail Bar
15           Cocktail Bar             Wine Bar        Ice Cream Shop
16               Wine Bar         Cocktail Bar   Japanese Restaurant
22                 Hotel           Restaurant                 Café
9                  Hotel    Italian Restaurant        Ice Cream Shop
0                  Hotel        Ice Cream Shop                Plaza

   5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
1    Monument / Landmark            Art Museum                Bakery
7          Sandwich Place            Art Museum                 Café
2     Italian Restaurant              Boutique   Monument / Landmark
3          Women's Store               Bakery           Men's Store
4               Boutique        Ice Cream Shop                Bakery
5                  Café           Restaurant    Seafood Restaurant
6    Japanese Restaurant                Plaza         Cocktail Bar
8     Seafood Restaurant                Plaza   Japanese Restaurant
23           Pizza Place        Ice Cream Shop             Bookstore
10      Sushi Restaurant           Pizza Place               Theater
11   Japanese Restaurant            Art Museum      Sushi Restaurant
12            Restaurant           Pizza Place        Clothing Store
```

13	Boutique	Cocktail Bar	Café
14	Japanese Restaurant	Café	Park
15	Café	Pizza Place	Restaurant
16	Pizza Place	Café	Ice Cream Shop
22	Ice Cream Shop	Japanese Restaurant	Cocktail Bar
9	Café	Monument / Landmark	Gift Shop
0	Monument / Landmark	Café	Art Museum

	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
1	Gift Shop	Coffee Shop	Bar
7	Gift Shop	Monument / Landmark	Theater
2	Park	Coffee Shop	Pizza Place
3	Monument / Landmark	Cocktail Bar	Lounge
4	Park	Seafood Restaurant	Art Gallery
5	Hotel	Pub	Cocktail Bar
6	Dessert Shop	Restaurant	Salon / Barbershop
8	Park	Shoe Store	Bookstore
23	Sandwich Place	Trattoria/Osteria	Szechuan Restaurant
10	Seafood Restaurant	Chinese Restaurant	Café
11	Café	Bakery	Pizza Place
12	Plaza	Café	Gift Shop
13	Bakery	Vegetarian / Vegan Restaurant	Art Gallery
14	Men's Store	Women's Store	Art Museum
15	Japanese Restaurant	Hotel	Bar
16	Theater	Restaurant	Bar
22	Wine Bar	Art Gallery	Gourmet Shop
9	Wine Bar	Theater	Bakery
0	Dessert Shop	Gift Shop	Pizza Place

2.15.9 Representing the top 10 venues for each cluster

First of all we have to associate the borough code and the related venue categories with the cluster label.

[75]:
```
neighborhoods_clusters_venues = pd.merge(milan_venues,␣
↪neighborhoods_venues_sorted[['Borough code','Cluster Labels',␣
↪'Neighborhoods']], on='Neighborhoods')
neighborhoods_clusters_venues = neighborhoods_clusters_venues.drop(['Latitude',␣
↪'Longitude', 'Venue_Name', 'Venue_Latitude', 'Venue_Longitude'], axis=1)
neighborhoods_clusters_venues.head()
```

[75]:

	Neighborhoods	Venue_Category	Borough code	Cluster Labels
0	DUOMO	Plaza	B12	1
1	DUOMO	Monument / Landmark	B12	1
2	DUOMO	Hotel	B12	1
3	DUOMO	Scenic Lookout	B12	1

```
4          DUOMO          Art Gallery          B12                 1
```

Now, we find how many times each venue category occurs.

[76]:
```
neighborhoods_clusters = neighborhoods_clusters_venues.groupby(by=['Cluster␣
↪Labels', 'Venue_Category'])['Venue_Category'].count()
neighborhoods_clusters = neighborhoods_clusters.to_frame(name='Frequency').
↪reset_index()
neighborhoods_clusters.sort_values(['Cluster Labels', 'Frequency'],␣
↪ascending=[True, False]).head()
```

[76]:
```
     Cluster Labels      Venue_Category  Frequency
140               0  Italian Restaurant        443
195               0         Pizza Place        326
45                0               Café        286
131               0               Hotel        159
134               0      Ice Cream Shop        148
```

and then we calculate the % for each venue category of both clusters.

[77]:
```
neighborhoods_clusters_0 =␣
↪neighborhoods_clusters[neighborhoods_clusters['Cluster Labels']==0]
neighborhoods_clusters_0 ['%'] = round(neighborhoods_clusters_0['Frequency']/
↪neighborhoods_clusters_0['Frequency'].sum()*100,1)
neighborhoods_clusters_0 = neighborhoods_clusters_0.sort_values('%',␣
↪ascending=False)
neighborhoods_clusters_0 = neighborhoods_clusters_0.reset_index(drop=True).
↪head(10)
neighborhoods_clusters_0 = neighborhoods_clusters_0.drop(['Cluster Labels',␣
↪'Frequency'], 1)
neighborhoods_clusters_0
```

```
/Users/segatto/opt/anaconda3/envs/Data_Science_env/lib/python3.6/site-
packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

[77]:
```
        Venue_Category    %
0   Italian Restaurant  9.4
1         Pizza Place  6.9
2               Café  6.1
3               Hotel  3.4
4      Ice Cream Shop  3.1
```

```
5          Supermarket  2.9
6           Restaurant  2.5
7         Cocktail Bar  2.5
8               Plaza  2.3
9  Japanese Restaurant  2.2
```

[78]:
```
neighborhoods_clusters_1 =␣
↪neighborhoods_clusters[neighborhoods_clusters['Cluster Labels']==1]
neighborhoods_clusters_1 ['%'] = round(neighborhoods_clusters_1['Frequency']/
↪neighborhoods_clusters_1['Frequency'].sum()*100,1)
neighborhoods_clusters_1 = neighborhoods_clusters_1.sort_values('%',␣
↪ascending=False)
neighborhoods_clusters_1 = neighborhoods_clusters_1.reset_index(drop=True).
↪head(10)
neighborhoods_clusters_1.drop(['Cluster Labels', 'Frequency'], 1)
```

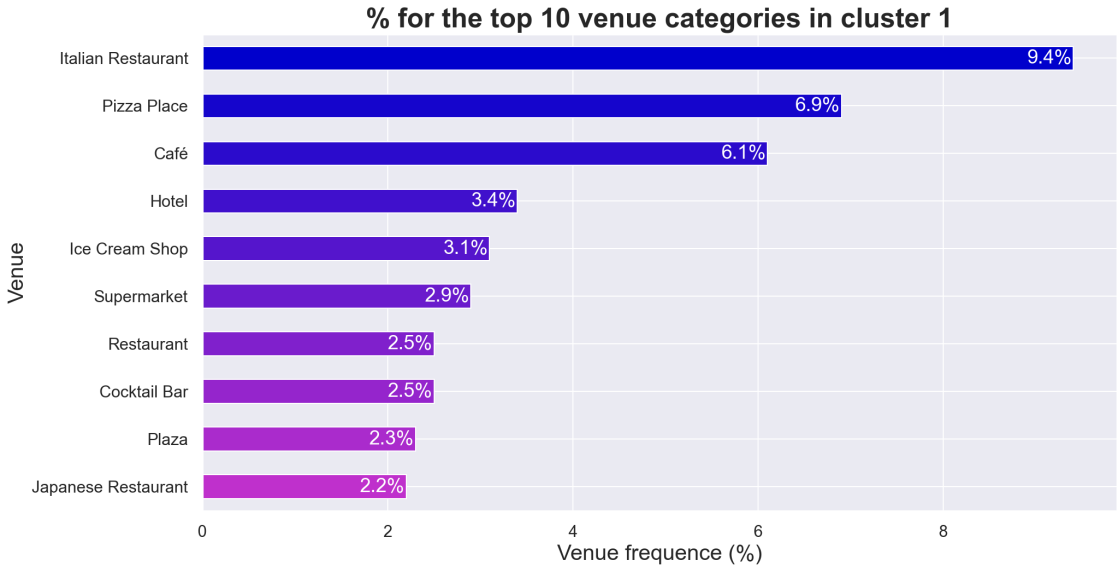
```
/Users/segatto/opt/anaconda3/envs/Data_Science_env/lib/python3.6/site-
packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

[78]:
```
        Venue_Category    %
0   Italian Restaurant  9.8
1               Hotel  6.4
2       Ice Cream Shop  5.2
3               Plaza  4.3
4                Café  3.7
5            Boutique  3.6
6         Pizza Place  3.1
7        Cocktail Bar  2.8
8  Japanese Restaurant  2.4
9              Bakery  2.1
```

now we can create the barh plots.

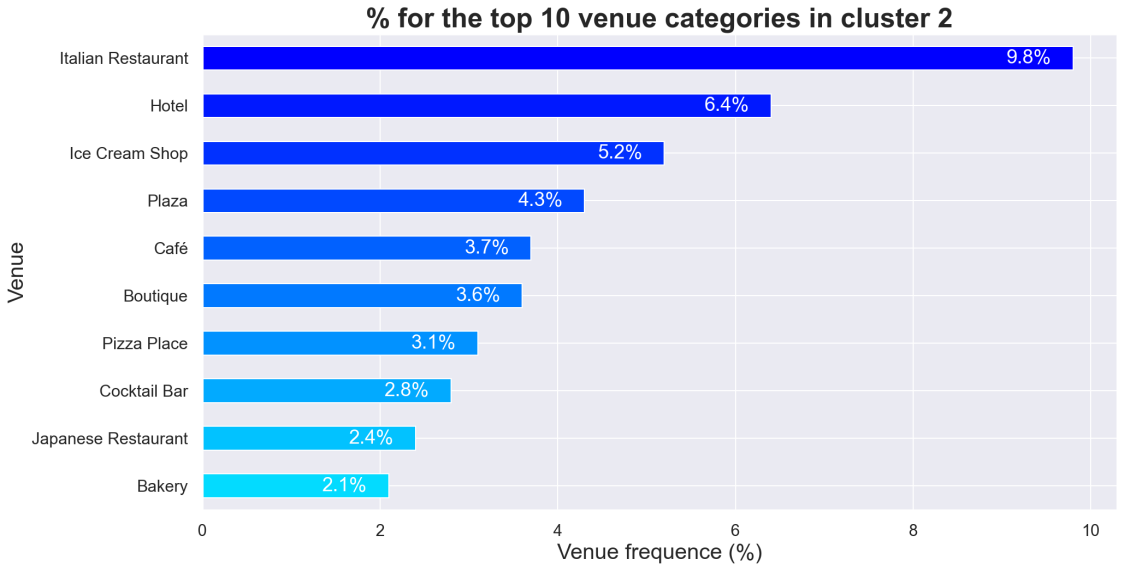
[79]:
```
customcmap = [(x/12.0,  x/48.0, 0.8) for x in␣
↪range(len(neighborhoods_clusters_0))]

ax = neighborhoods_clusters_0.plot(kind='barh', x='Venue_Category', y='%',␣
↪color=customcmap, figsize=(15, 8), fontsize=15, legend=False)
plt.xlabel('Venue frequence (%)', fontsize=20)
plt.ylabel('Venue', fontsize=20)
```

```
ttl = plt.title('% for the top 10 venue categories in cluster 1', fontsize=25,␣
 ↪weight='bold')
ttl.set_position([.5, 1.06])
ax.invert_yaxis()

ctb = LinearSegmentedColormap.from_list('custombar', customcmap, N=2048)
sm = plt.cm.ScalarMappable(cmap=ctb)

for i, (p, pr) in enumerate(zip( neighborhoods_clusters_0['Venue_Category'],␣
 ↪neighborhoods_clusters_0["%"])):
    plt.text(s=str(pr)+"%", x=pr-0.5, y=i,␣
 ↪color="w",verticalalignment="center", horizontalalignment="left", size=18)
```



[80]:
```
customcmap = [(x/900.5,  x/10.5, 1) for x in␣
 ↪range(len(neighborhoods_clusters_1))]

ax = neighborhoods_clusters_1.plot(kind='barh', x='Venue_Category', y='%',␣
 ↪color=customcmap, figsize=(15, 8), fontsize=15, legend=False)
plt.xlabel('Venue frequence (%)', fontsize=20)
plt.ylabel('Venue', fontsize=20)
ttl = plt.title('% for the top 10 venue categories in cluster 2', fontsize=25,␣
 ↪weight='bold')
ttl.set_position([.5, 1.06])
ax.invert_yaxis()

ctb = LinearSegmentedColormap.from_list('custombar', customcmap, N=2048)
sm = plt.cm.ScalarMappable(cmap=ctb)
```

```
for i, (p, pr) in enumerate(zip( neighborhoods_clusters_1['Venue_Category'],␣
↪neighborhoods_clusters_1["%"])):
    plt.text(s=str(pr)+"%", x=pr-0.75, y=i,␣
↪color="w",verticalalignment="center", horizontalalignment="left", size=18)
```



2.15.10 Calculate the average buy/rent prices of each cluster

2.15.11 Cluster 0

[81]:
```
neighborhoods_venues_sorted.loc[neighborhoods_venues_sorted['Cluster␣
↪Labels']==0,['Buy_price','Rent_price']].mean(axis=0)
```

[81]:
```
Buy_price     3123.417722
Rent_price      10.065823
dtype: float64
```

2.15.12 Cluster 1

[82]:
```
neighborhoods_venues_sorted.loc[neighborhoods_venues_sorted['Cluster␣
↪Labels']==1,['Buy_price','Rent_price']].mean(axis=0)
```

[82]:
```
Buy_price     8512.719298
Rent_price      26.168421
dtype: float64
```

3 4. Discussion

This analysis revealed some important aspects to take into account when someone decides to buy or to rent a house in Milan and they don't know in which neighborhood go to live. First of all, the house price is one the main factors that affects the house choice. It's good to know which Milan areas have house prices in line with the budget of the subject: this will allow him to refine its choice and optimize the time of house research. For example, a person who intends to invest a large sum of money could focus to the house for sale/rent localized in B and C areas (the most expensive but, obviously, the most prestigious ones). On the contrary, a person that plans to spend relatively small amount of money can focus on the house located in D and E areas (the most distant to the city center but, at the same time, the most affordable ones). However, it's important to considerate the state of the house: indeed, a house located in the same Milan areas but in different condition will have a different price. Certainly, a stately house is dearer that a residential house in excellent condition; this last one, in turn, is less convenient that a residential house in normal condition. It's better to have a clear idea about the condition of the house that it's looking for, because this aspect may greatly affect the house value. However, if this is not enough and someone wants to focus the attention on a smaller group of choice, they could consider the distance beyond which the price is no longer in line with the predefined budget, this will allow them to select the more aligned neighborhoods. In general, there is a rapid decrease of the market/rental values of the residential house, until a distance of 4 km from the city center, then the price decrease more slowly (beyond 4 km only neighborhoods belonging to D and E areas are present). Different situation occurs for the stately houses, indeed the values decrease in a more uniform way. Someone who want to buy or to rent a house in Milan would do well to choose the best neighborhoods to live also according to their interests. This will ensure less use of the car and the public transport, therefore less stress. This study has found out that Milan neighborhoods can be split in 2 clusters: the first one includes the neighborhoods closer to the city center and the second one the neighborhoods located in the suburbs. It has been observed that cluster 1 is rich in hotels, attractive plazas, boutiques, monuments, Italian restaurants, art galleries, etc.: all venues that attract tourists and people who love art. Cluster 2, on the other hand, includes many places that serving food and drinks, in addition to park, hotels, etc. The neighborhoods belonging to this cluster are the perfect ones for people that live in Milan and don't give up enjoying social life. Finally, it is possible to notice that in general Milan, even if it is a metropolis, is rich in parks. This is an important aspect, since it's able to offer enjoyment to those who appreciate outdoors life and for people who have dogs.

What could be done better? This project only considers 2 aspects that could affect the neighborhood choice i.e. house value and the proximity to the venues of interests. However, there are many more characteristics that may come into play, like the presence of public transports (it's well known that traffic on the roads of big city is particularly heavy) or the house characteristics such as number of rooms, bathrooms etc. Future research could develop a methodology that allows to consider more aspects. In addition, this project has been made by the use of Foursquare API Sandbox Tier Developer account that has limitation on the number of API call (in this case 100 for each neighborhood), this could have been affected the real venue frequencies. In the future this problem may be bypassed using a paid account. Finally, it would be interesting to split the Milan neighborhoods in more than 2 clusters, as to improve the differentiation of the neighborhoods according to the venue frequencies. However, this would affect the K-means precision.

4 5. Conclusion

This project has been intended as a tool to help people who intend to buy or to rent a house in Milan, as to facilitate the choice concerning the place where focus the house research. First of all, data has been collected from the Italian Revenue Agency website, all the neighborhoods have been geolocated (by using Google Maps Geocoding API) and then mapped to determinate the exact position. Subsequently, the data were analyzed, as to determinate the correlation between the house location (Milan area and neighborhood) and the related price, also depending on the state of the home, and the most frequent venues that are present (by using Foursquare API). It has been possible, on the whole, to find satisfactory answers to the question cornerstone of this study. Indeed, based on these findings, I can conclude that there is a positive correlation between the house prices and the proximity to the city center; moreover, it has been possible to split the neighborhoods in different clusters, depending on the venue frequencies, and so determine the differences among them. Obviously, there may be other variables at play that also contribute to the house choice. However, it is something that could help someone during the decision process of the future house location.

[ ]: