In [1]:

```python
import pandas as pd
import numpy as np
import geohash as gh

import scipy as sp
from statistics import median

import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import seaborn as sns

import os
```

In [2]:

```python
from bokeh.io import output_file, output_notebook, show

from bokeh.models import (
  GMapPlot, GMapOptions, ColumnDataSource, Circle, LogColorMapper, BasicTicker, ColorBar,
    Range1d, PanTool, WheelZoomTool, BoxSelectTool)
from bokeh.models.mappers import ColorMapper, LinearColorMapper
from bokeh.palettes import Viridis5
from bokeh.plotting import gmap
from bokeh.transform import linear_cmap
from bokeh.palettes import Plasma256 as palette
```

In [3]:

```python
apartments = pd.read_csv("dataset.csv", delimiter=',')
```

In [4]:

```python
apartments.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 868 entries, 0 to 867
Data columns (total 4 columns):
longitude         868 non-null float64
latitude          868 non-null float64
price_sqm         868 non-null int64
living_area_sqm   868 non-null float64
dtypes: float64(3), int64(1)
memory usage: 27.2 KB
```

In [5]:

```python
plt.style.use('seaborn-deep')
x = apartments['price_sqm']
bins = np.linspace(2000, 10000, 100)
plt.hist([x], bins, label =['Price sqm'])
plt.title('Distribution of price_sqm')
plt.xlabel('Euros')
plt.ylabel('Appartments')
plt.legend(loc='upper right')
plt.show()
```
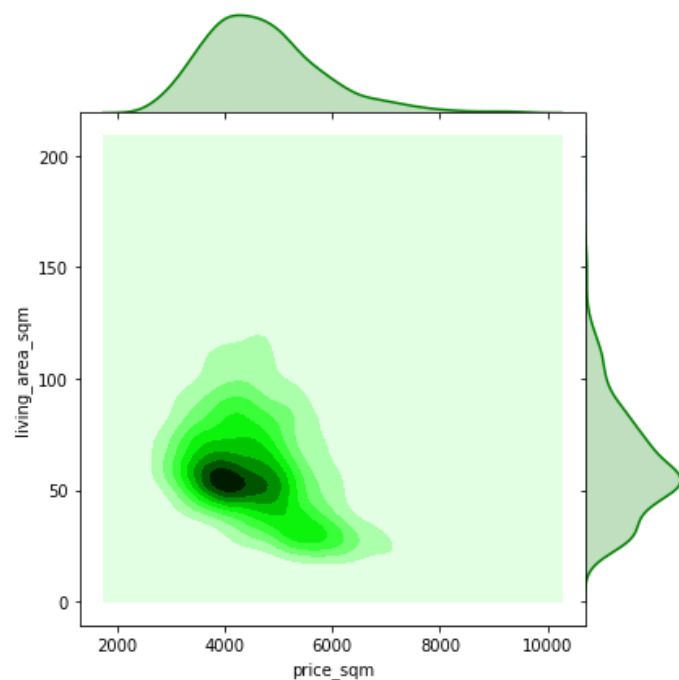

In [6]:
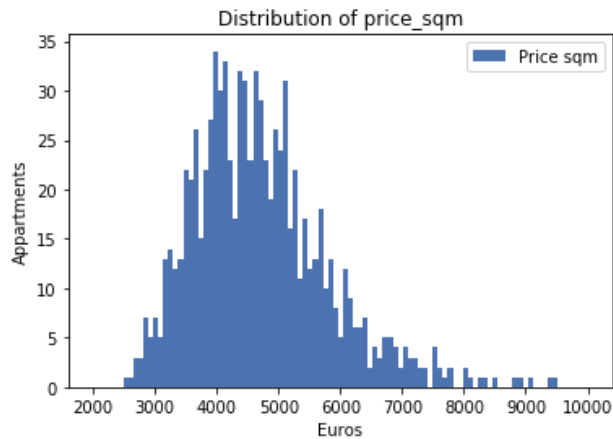
```python
sns.jointplot("price_sqm", "living_area_sqm", apartments, kind="kde", space=0, color="g")
```

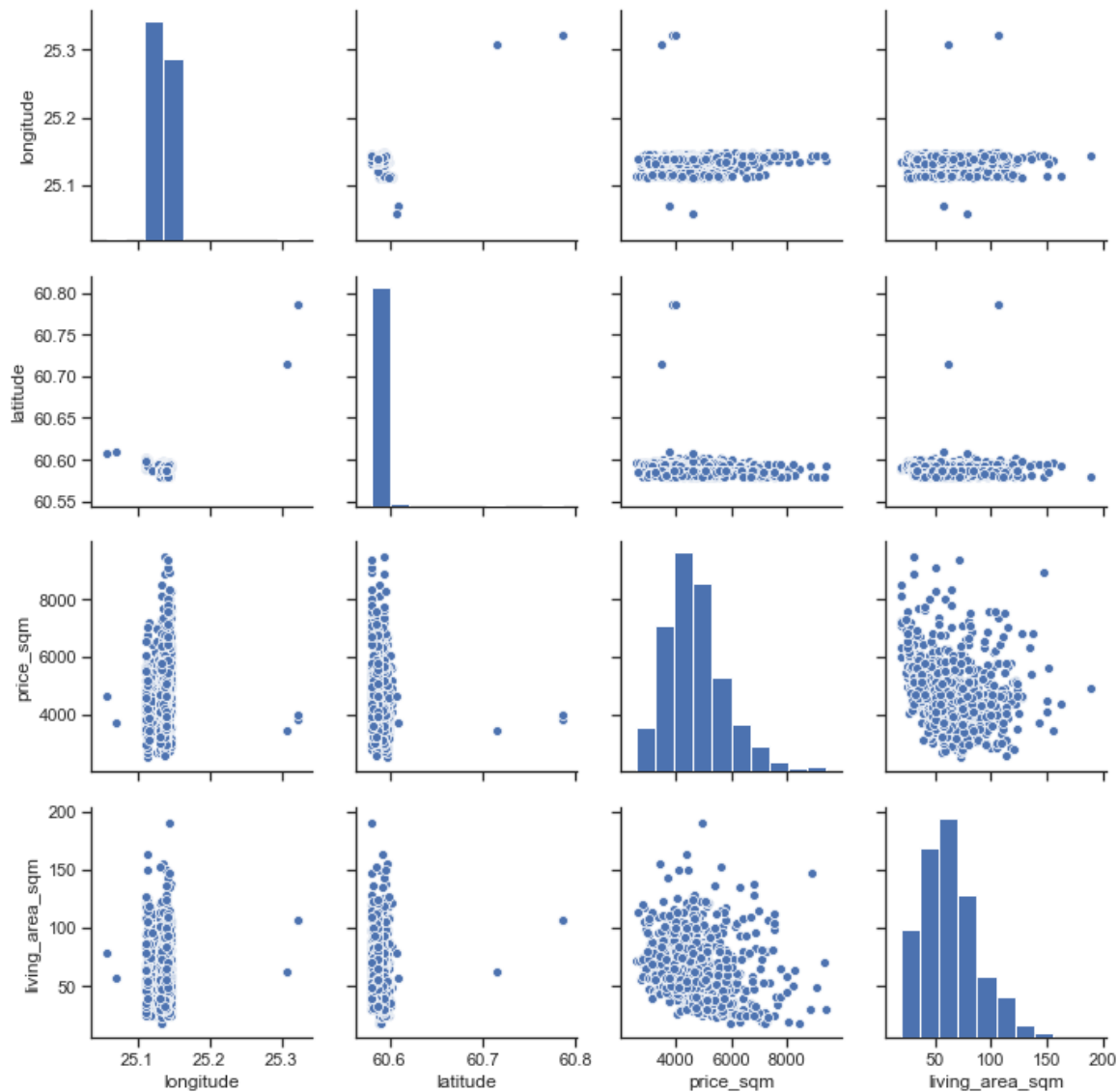Out[6]:

```
<seaborn.axisgrid.JointGrid at 0x252a9bb9888>
```

In [7]:

```
sns.set(style="ticks", color_codes=True)
g= sns.pairplot(apartments)
```


In [8]:

```
apartments['geohash']=apartments.apply(lambda x: gh.encode(x.latitude, x.longitude, precision=9), axis=1)
apartments['district_hash']=apartments.apply(lambda x: gh.encode(x.latitude, x.longitude, precision=7), axis=1)
```

In [9]:

```
apartments.head()
```

Out[9]:

	longitude	latitude	price_sqm	living_area_sqm	geohash	district_hash
0	25.12542	60.5894	3472	70.0	udcbetzyb	udcbetz
1	25.12542	60.5894	5726	24.0	udcbetzyb	udcbetz
2	25.12542	60.5894	4443	35.0	udcbetzyb	udcbetz
3	25.12542	60.5894	3518	72.0	udcbetzyb	udcbetz
4	25.12542	60.5894	3976	60.0	udcbetzyb	udcbetz

In [10]:

```python
data_dict = {}
for g in apartments['geohash']:
    data_dict[g] = [apartments[apartments['geohash']==g]['latitude'].tolist()[0],apartments[apartments['geohash']
                    apartments[apartments['geohash']==g]['geohash'].tolist()[0],apartments[apartments['geohash']=
                    np.median(apartments[apartments['geohash']==g]['price_sqm']),sum(apartments[apartments['geoha
```

In [11]:

```python
buildings = pd.DataFrame.from_dict(data_dict, orient='index',columns=['latitude', 'longitude','geohash','distric
```

In [12]:

```python
buildings
```

Out[12]:

	latitude	longitude	geohash	district_hash	median_price_sqm	sqm
udcbetzyb	60.589400	25.125420	udcbetzyb	udcbetz	3976.0	261.0
udcbsnk5u	60.591657	25.142393	udcbsnk5u	udcbsnk	4740.0	435.0
udcbevbf2	60.588642	25.126783	udcbevbf2	udcbevb	5767.5	154.5
udcbey2wn	60.592024	25.126703	udcbey2wn	udcbey2	3838.5	335.0
udcbetzm2	60.589169	25.124740	udcbetzm2	udcbetz	3637.0	361.0
...	...	...	...	...	...	...
udcbsj5f5	60.584474	25.142022	udcbsj5f5	udcbsj5	5482.0	112.0
udcbevw15	60.587037	25.134114	udcbevw15	udcbevw	5691.0	153.0
udcbsj2m4	60.586365	25.137170	udcbsj2m4	udcbsj2	6758.0	54.5
udcbsh7by	60.580142	25.142167	udcbsh7by	udcbsh7	7890.0	338.0
udcbsjd4f	60.587356	25.139558	udcbsjd4f	udcbsjd	4370.0	327.5

299 rows × 6 columns

In [13]:

```python
map_data =buildings.copy(deep=True)
map_data['sqm'] = map_data['sqm']/50
```

In [14]:

```python
map_options = GMapOptions(lat=60.59, lng=25.13, map_type="roadmap", zoom=13)
p = gmap("AIzaSyAaVKvTSX6yfFp0KPUdbIZ6sjFMAR7RMc0", map_options, title='Kodit data',
         width=800, height=600)
```

In [15]:

```python
source = ColumnDataSource(map_data)
```

In [16]:

```python
color_mapper = LinearColorMapper(palette=Viridis5)

mapper = linear_cmap('median_price_sqm', palette, 2500., 9500.)

p.circle('longitude', 'latitude',size='sqm', alpha=0.2,
                    color=mapper, source=source)

color_bar = ColorBar(color_mapper=mapper['transform'],
                     location=(0,0))
p.add_layout(color_bar, 'right')
output_notebook()
```

BokehJS 1.3.4 successfully loaded.
(https://bokeh.pydata.org)

In [17]:

```
show(p)
```

Kodit data
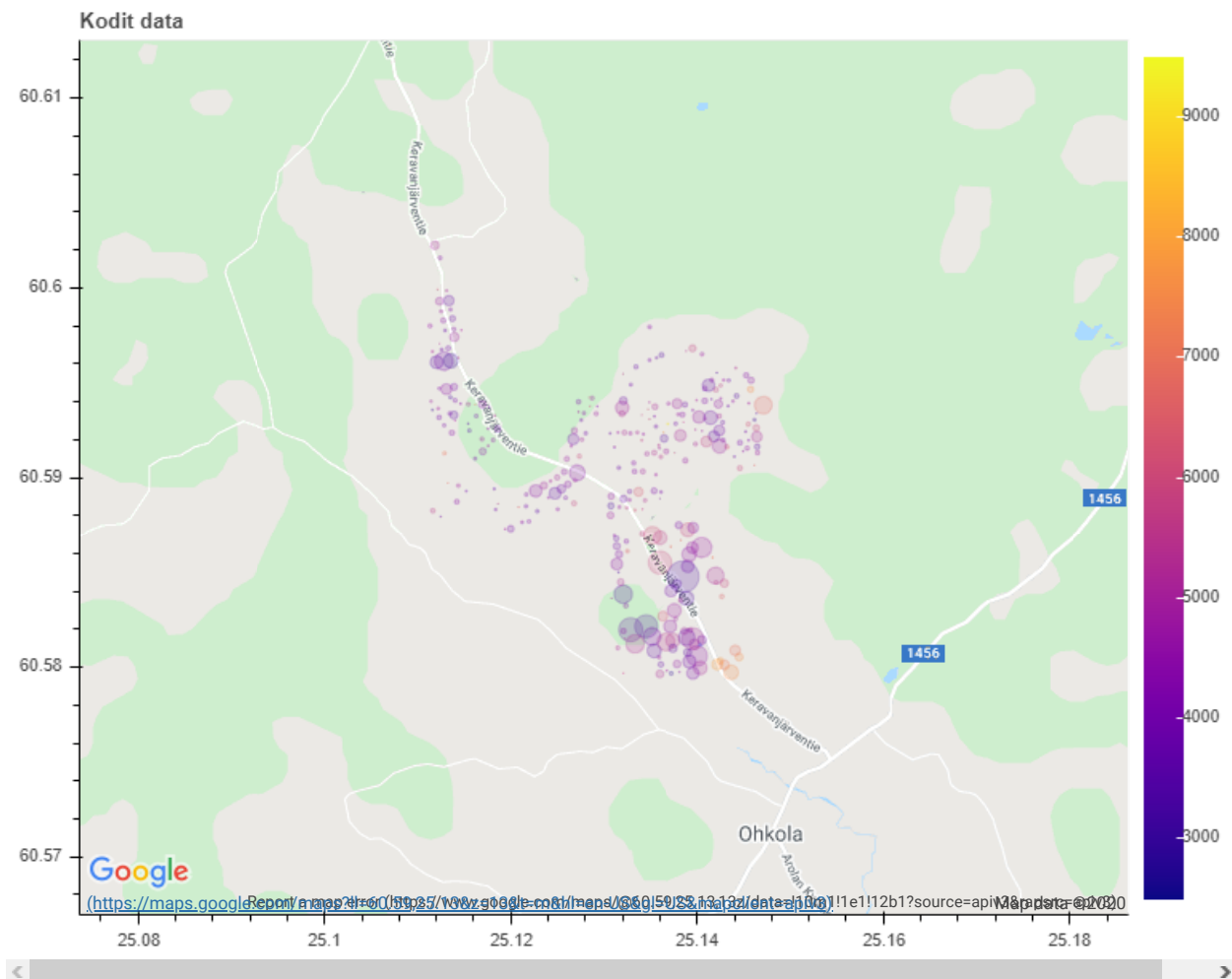
In [18]:

```
price_range = 2000
hom_distr_counter = 0
mixed_distr_counter = 0
```

In [19]:

```
for d in buildings['district_hash']:
    max_price_sqm = max(buildings[buildings['district_hash']==d]['median_price_sqm'])
    min_price_sqm = min(buildings[buildings['district_hash']==d]['median_price_sqm'])

    if (max_price_sqm-min_price_sqm) < price_range:
        buildings.loc[buildings['district_hash'] ==d, 'homogeneous'] = 1
        hom_distr_counter = hom_distr_counter + 1
    else :
        buildings.loc[buildings['district_hash'] ==d, 'homogeneous'] = 0
        mixed_distr_counter = mixed_distr_counter + 1
```

In [20]:

```
buildings
```

Out[20]:

	latitude	longitude	geohash	district_hash	median_price_sqm	sqm	homogeneous
udcbetzyb	60.589400	25.125420	udcbetzyb	udcbetz	3976.0	261.0	1.0
udcbsnk5u	60.591657	25.142393	udcbsnk5u	udcbsnk	4740.0	435.0	0.0
udcbevbf2	60.588642	25.126783	udcbevbf2	udcbevb	5767.5	154.5	1.0
udcbey2wn	60.592024	25.126703	udcbey2wn	udcbey2	3838.5	335.0	1.0
udcbetzm2	60.589169	25.124740	udcbetzm2	udcbetz	3637.0	361.0	1.0
...	...	...	...	...	...	...	...
udcbsj5f5	60.584474	25.142022	udcbsj5f5	udcbsj5	5482.0	112.0	1.0
udcbevw15	60.587037	25.134114	udcbevw15	udcbevw	5691.0	153.0	1.0
udcbsj2m4	60.586365	25.137170	udcbsj2m4	udcbsj2	6758.0	54.5	1.0
udcbsh7by	60.580142	25.142167	udcbsh7by	udcbsh7	7890.0	338.0	1.0
udcbsjd4f	60.587356	25.139558	udcbsjd4f	udcbsjd	4370.0	327.5	1.0

299 rows × 7 columns

In [21]:

```
mixed_distr_counter
```

Out[21]:

44

In [ ]:

In [ ]:

In [ ]: