

St. Gallen Real Estate Rental Market Visualization

May 21, 2019

University of St. Gallen

A: Skills: Programming with Advanced Computer Languages

B: Skills: Programming: Introduction Level

Authors - Student ID - Course

Leonardo Aldeggi - 18-605-303 - A

Dominik Feuz - 13-607-163 - A

Daphne Huang - 16-610-784 - B

Camilla Weder - 16-620-809 - B

1 Project Description

This is a student project of the University of St. Gallen of the courses *Programming: Introduction Level* and *Programming with Advanced Computer Languages*. In order to find a suitable and reasonably priced apartment for the students of the University of St. Gallen, the aim of this project is to program a map displaying the rental housing market situation in St. Gallen. The various properties are to be sorted according to their price level. First, this will enable the most efficient way of finding a suitable apartment in St. Gallen. Second the code can be modified slightly, and a new dataset can be added to represent any desired area. With this project it is possible to sort and display the prices of any housing market in a specific geographical region. This facilitates the decision-making process of finding the right rental apartment.

In order to achieve the set objective, we collect and filter data of the real estate market in St. Gallen. For this purpose, we collect data by web scraping Homegate.ch through a Google Chrome extension, "Data Scraper - Easy Web Scraping", to extract data out of HTML web pages and import it into Microsoft Excel spreadsheets. We clean up the data and copy it in a .txt file, together with GPS coordinates.

We then import the data with pandas and display the data using Folium, a Python library that helps create Leaflet maps. We initially planned to use Google Maps, but we then discovered that we would have needed to create an account registered with a payment method in order to get the API. Although there is a free credit of a \$200, we didn't want to take any risk of being billed. So, we opted for this open-source library. The final code using Folium is described in section 2.

2 Code Description

We are going to be using a package called folium, but we can't import it right away. We need to first install it. The way we do that is through the line of code below.

```
pip install folium
```

We will bring the dataset into python through the following: `pd.read_csv()`. We extracted the data using a web crawler tool from `homegate.ch` and manually added the geolocation data (i.e. Latitude and Longitude). We selected the relevant variables, in our case the price, the number of rooms, the size of the apartment, and the address.

```
import folium
import pandas as pd
import numpy as np
data=pd.read_csv(" StGallen_RealEstate_Project.txt ",skiprows=10, header=None, names =
["Price","Rooms","Size","Address","Street_number","LAT","LON"])
print(data)
```

Now we clean the data input and standardize it by converting into numeric format.

```
# In[3]:
data["LAT"] = pd.to_numeric(data["LAT"])
data["LON"] = pd.to_numeric(data["LON"])
data["Price"] = pd.to_numeric(data["Price"])
data["Rooms"] = pd.to_numeric(data["Rooms"])
data.info()
```

We define the characteristics of the map that we intend to display as a result. Averaging the latitude as well as the longitude data returns a specific point on the map to be displayed. We define the zoom factor and the type of map that we intend to display (i.e. `OpenStreetMap`).

```
# In[4]:
map=folium.Map(location=[data['LAT'].mean(),data['LON'].mean()],zoom_start=14,tiles='OpenStreetMap')
```

It is our objective to categorize and visualize each available apartment on the map. To facilitate the viewer experience, we differentiate the color given the price of the apartment. This differentiation does not capture differences in the number of rooms. This could be part of an additional project. Our sole objective is to give a first overview of the price patterns in the city. For this purpose, we defined 3 price categories. The lowest one ranges from 0 to 1000 and is indicated with green, the middle one from 1001 to 1400 and is indicated with orange as well as the highest and last category which is indicated with red and goes up from 1401.

```
# In[5]:  
def color(price):  
    if price in range(0,1000):  
        col='green'  
    elif price in range(1001,1400):  
        col='orange'  
    else:  
        col='red'  
    return col
```

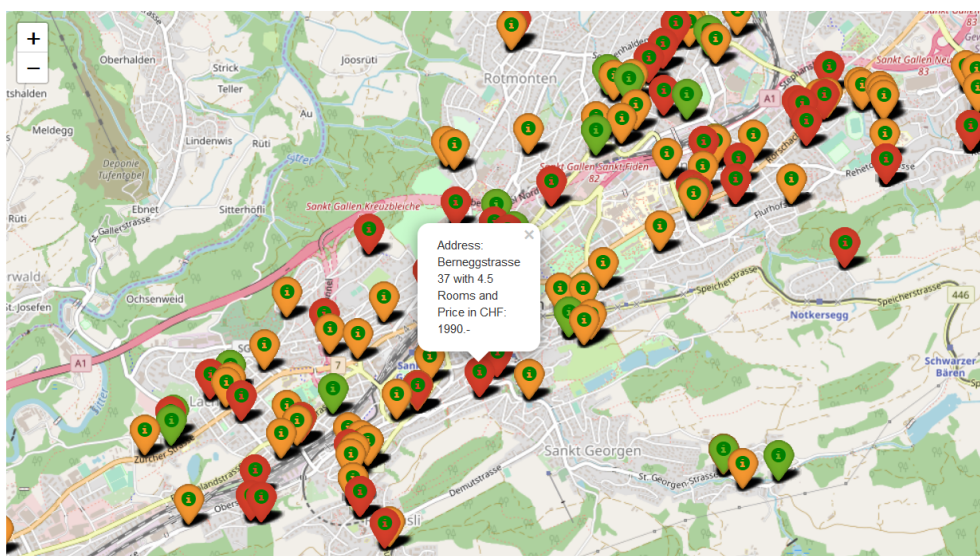
Building on the folium package syntax, we attach a popup to each marker displaying the address, the number of rooms and the price. This way, the viewer can focus on the area of his or her interest and compare the price level with the size of the respective apartment.

```
# In[50]:
fg = folium.FeatureGroup(name="St. Gallen Real Estate Market")
for lat,lon,Rooms,Price,Address,Street_number in
zip(data['LAT'],data['LON'],data['Rooms'],data['Price'],data['Address'],data['Street_number']):
    fg.add_child(folium.Marker(location=[lat,lon],popup="Address: "+ Address +" "+ str(Street_number)
+ " " + "with "+ str(Rooms)+ " Rooms " + "and Price in CHF: "+ str(Price)+ "-
",icon=folium.Icon(color=color(Price),icon_color='green'))))
```

Last, we print the map in accordance with the predefined parameters.

```
#In[51]:
map.add_child(fg)
```

Output:



Additional comment: to save the final map result, we can save it in a HTML file on the desktop.

```
#In[52]:
map.save(outfile='/Users/daphnehuang/Desktop/StGallen RealEstate.html')
```