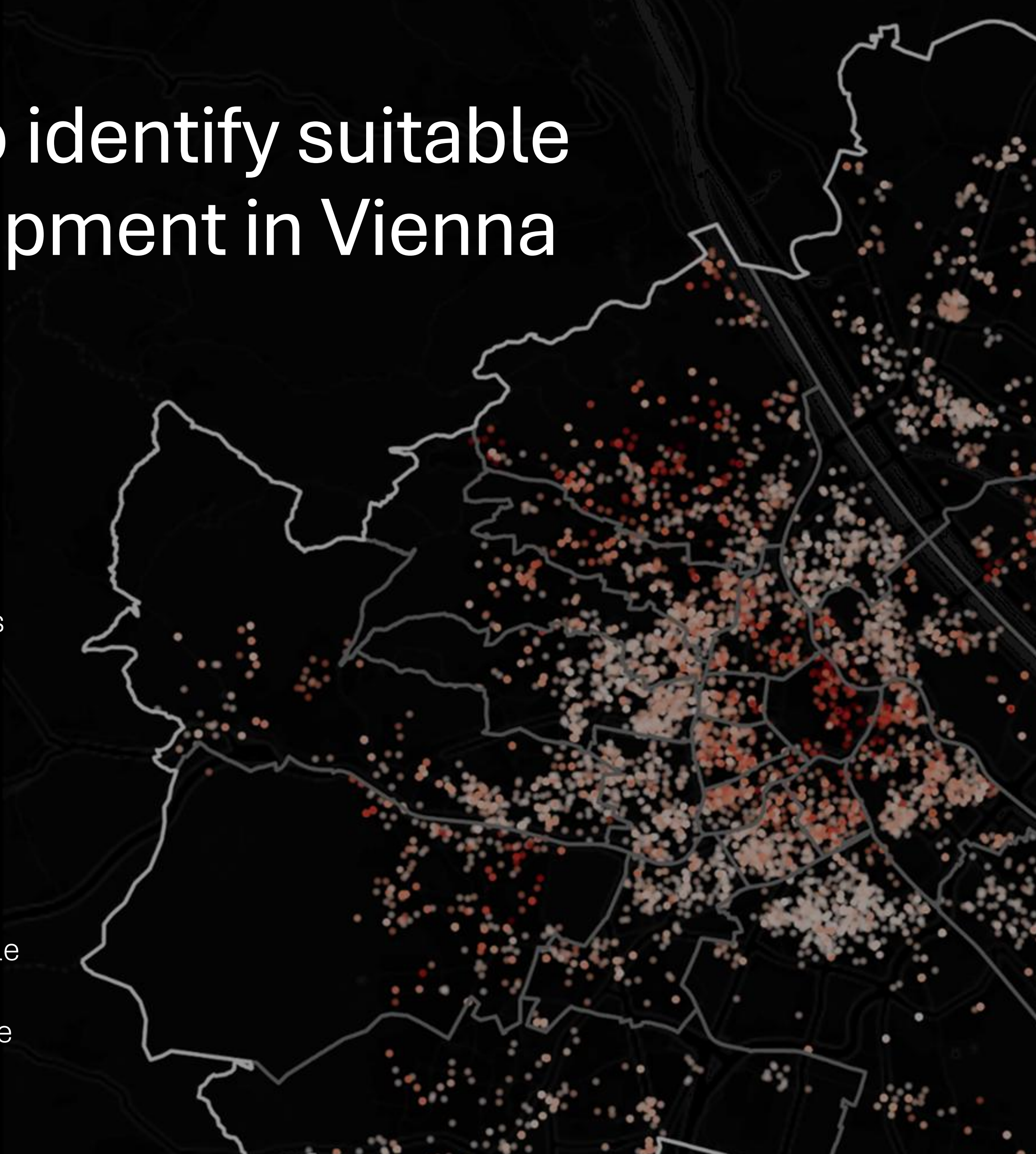
Using geospatial insights to identify suitable areas for real estate development in Vienna

Introduction to Geospatial Data Science

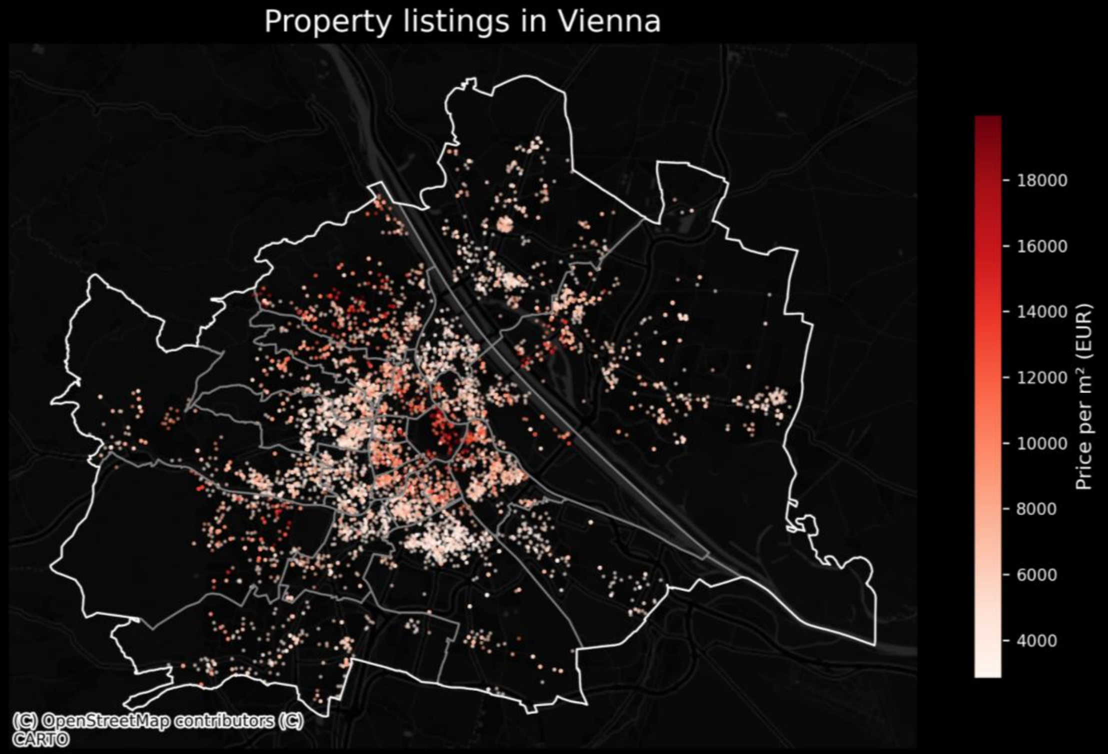
Author: Márton Nagy

Executive summary

Finding the most prospective areas for real estate development is a key task at real estate companies. However, this is most often done using expert knowledge rather than data-driven solutions. This report proposes a data-driven approach to identify these areas using for-sale real estate data enriched with different geospatial features. The goal is achieved through webscraping, geospatial data enrichment, predictive modeling, and then leveraging the model to predict property prices per m^2 all across Vienna. The predictions are then used to identify the most suitable areas for development in current residential areas, which are defined as having relatively high prices but low competition. In the end, 16 such areas are identified.



Real estate data on for-sale properties is nicely scrapable from WillHaben


Property listings in Vienna
18000
16000
14000
12000
10000
8000
6000
4000
Price per m² (EUR)
(C) OpenStreetMap contributors (C) CARTO

Data gathering

WillHaben.at is the largest online marketplace in Austria, which has a vast collection of properties for sale. Using the API powering the website, I scraped 16,981 Viennese listings in total on 17 March 2025. The scraped data includes features about the

- price (from which I calculated the m^2 prices),
- location (e.g. floor, coordinates),
- size (e.g. area, rooms),
- advertiser,
- property type,
- and certain amenities (e.g. terrace size).

I added certain calculated features to the dataset , e.g. ad age, and dummies for new and renovated properties based on the textual description.

Data cleaning

Apart from the obvious data-wrangling tasks (like filtering duplicates and type conversions), I dropped observations with too extreme m^2 prices and unreasonably large properties, and imputed unreasonably large room counts with the median (and replaced zero-rooms with ones). I grouped infrequent advertiser IDs and property types into respective *other* categories. In the end, I ended up with 14,882 properties in my cleaned dataset (as shown on the figure).

Technical notes for maps

All maps plotted in this document use EPSG:31287, which is the local Austrian CRS in meters. Underlying basemaps are provided by Carto. The main colormap used is represented by the legend. Other features are indicated either in the title or in notes, if applicable.

Scraped data can be enriched from GHS raster data and from OSM

Raster enrichment from GHS

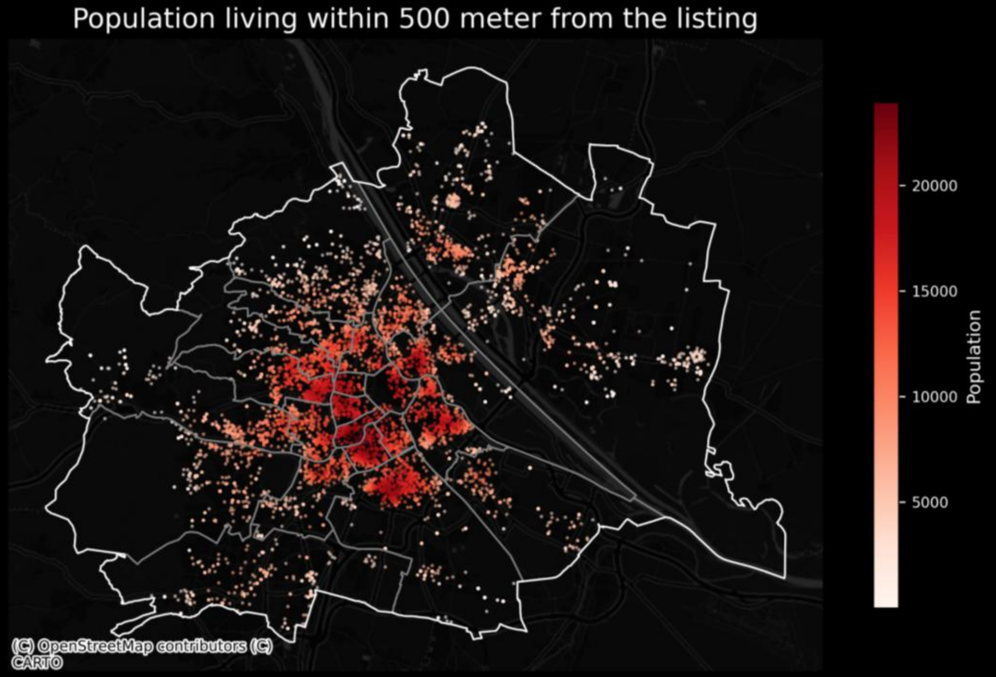
Raster data on population and built-in volume has been downloaded from GHS at 3 arcsec resolution, both for 2025 and 2030. Based on this, I calculated totals within a 500-meter radius of each listing, as well as expected growth in these areas for the two features. An example for this is visualized on the top map.

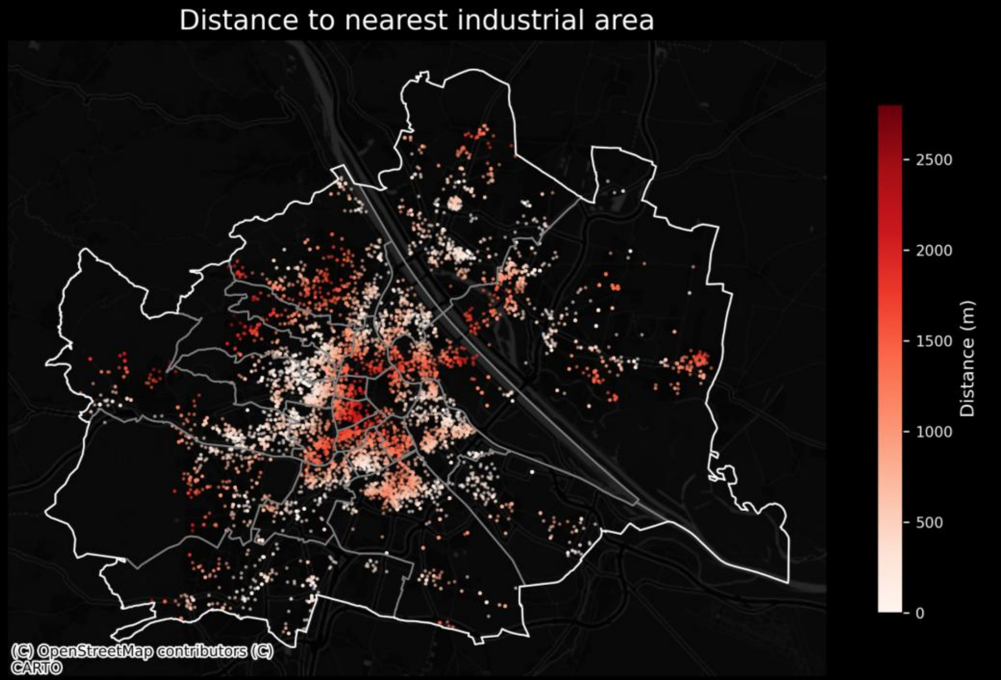
OpenStreetMap enrichment

I downloaded map features from OSM like different amenities, public transport options, different landuse categories or rivers. Then, for each listing, I calculated the closest instance and the count of instances within 100 (immediate proximity) and 800 (walking distance) meters. I did not calculate the counts for landuse categories and waterways as these would have been meaningless. An example for OSM enrichment is visualized in the bottom map.

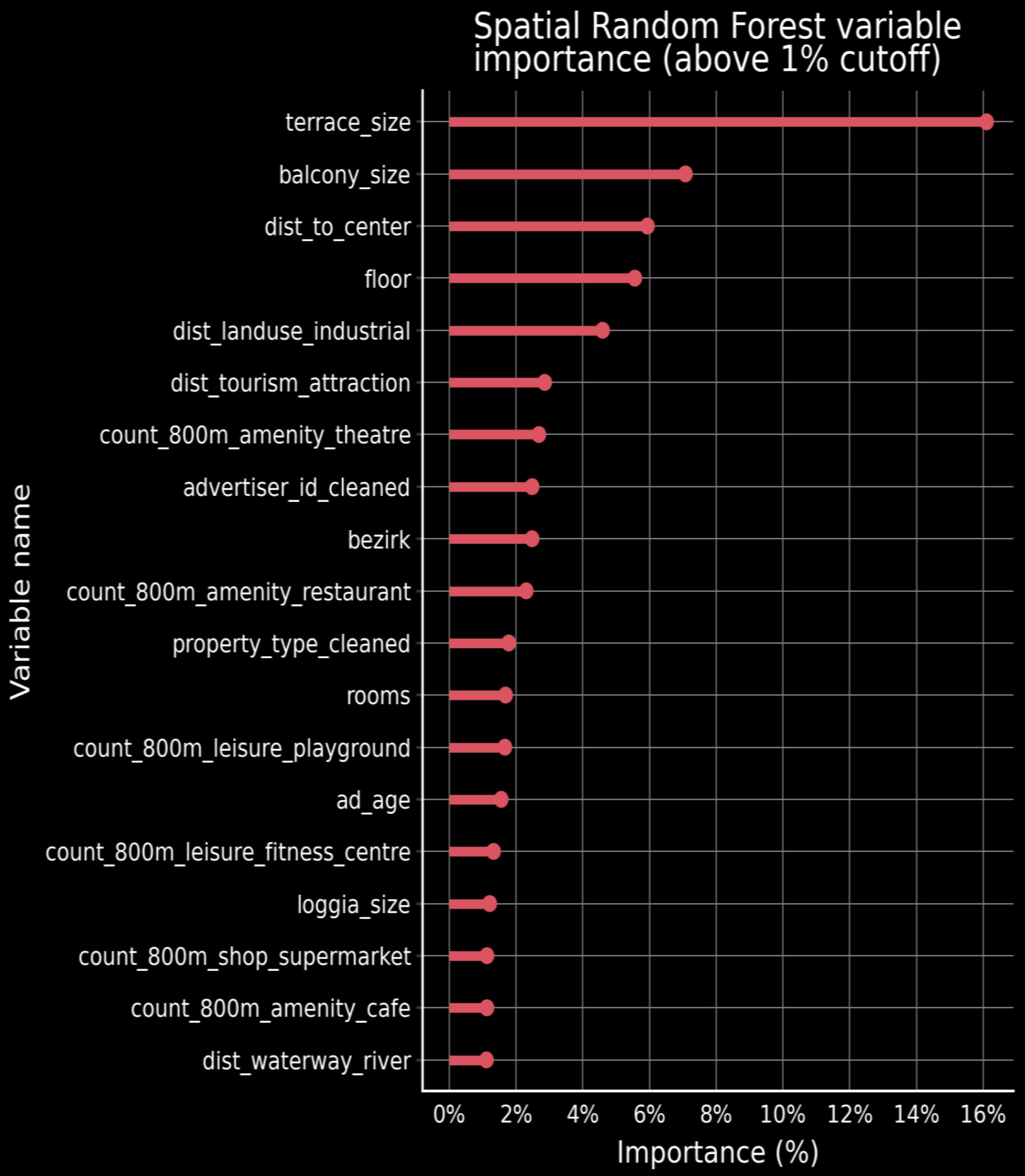
Other spatial feature generation

I also calculated the number of listings within 500 meters of each listing to get a measure for local competition. I identified the neighborhood each listing is located in using neighborhood shapes. I also added distance to the city center of Vienna. Note that I did not calculate local spatial correlation metrics, as my end goal is prediction, and correlations cannot be calculated out of sample.


Population living within 500 meter from the listing
20000
15000
10000
5000
Population
(C) OpenStreetMap contributors (C) CARTO


Distance to nearest industrial area
2500
2000
1500
1000
500
0
Distance (m)
(C) OpenStreetMap contributors (C) CARTO

As the end-goal was prediction, I opted for a Spatial Random Forest model



The choice of Spatial Random Forest

The use of traditional ML algorithms instead of true geospatial models was guided by the fact that the latter can hardly be used for out-of-sample predictions. SEM or SLM models rely on spatial weight matrices, so predictions for new observations can only be calculated on an iterative basis (computationally expensive). Also, as I had more than 100 input features, I could leverage ML models' ability to perform variable selection. Other models like boosting or neural networks could have been used instead of RF, but I deemed RF appropriate for such a pilot project.

Input features and cross-validation

All together, I had 116 input features (with one-hot-encoded categoricals). To tune the model, I used 5-fold cross-validation with a random search of 100 iterations over the parameter space (CV RMSE: 1330 or 19% of sample mean). The residuals' local spatial correlation was very close to zero, though statistically significant.

Model explanation

The figure tells us that the most important predictors are terrace and balcony size. However, spatial features are also quite important – like distance to the city center, distance to industrial sites, or to tourist attractions. For more details, refer to the SHAP plot included in the code.

Areas for development were identified through spatial shuffling and binning predicted prices

Spatial shuffling and predictions

Proporties’ price does not only depend on geospatial features, but also on more basic characteristics (like terrace size). Thus, a sample we want predictions for should also contain these features, but in a spatially independent way. This way, when multiple predictions are averaged over an area, we get mean predictions that are independent of the non-spatial features. To achieve this, I took the original data, concataneted the non-spatial features after each other 3-times (to have 3-times as many observations), but I assigned for each listing random coordinates inside Vienna. Then I enriched this set with geospatial features as well, and made the predictions, as illustrated on the top map.

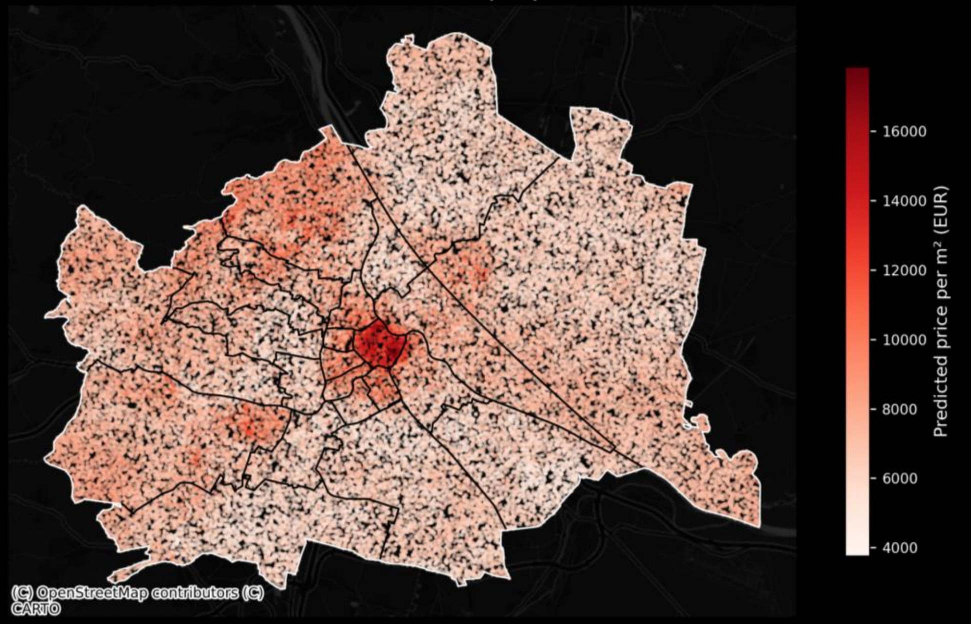
Spatial binning

The predictions were then averaged over an H3 hexagon grid covering Vienna. This way we got to the average price of the *average Viennese property* in each hexagon. On average, 76 predictions fell into one hexagon, which I deemed appropriate to consider non-spatial features averaged out.

Identifying suitable areas for real estate development

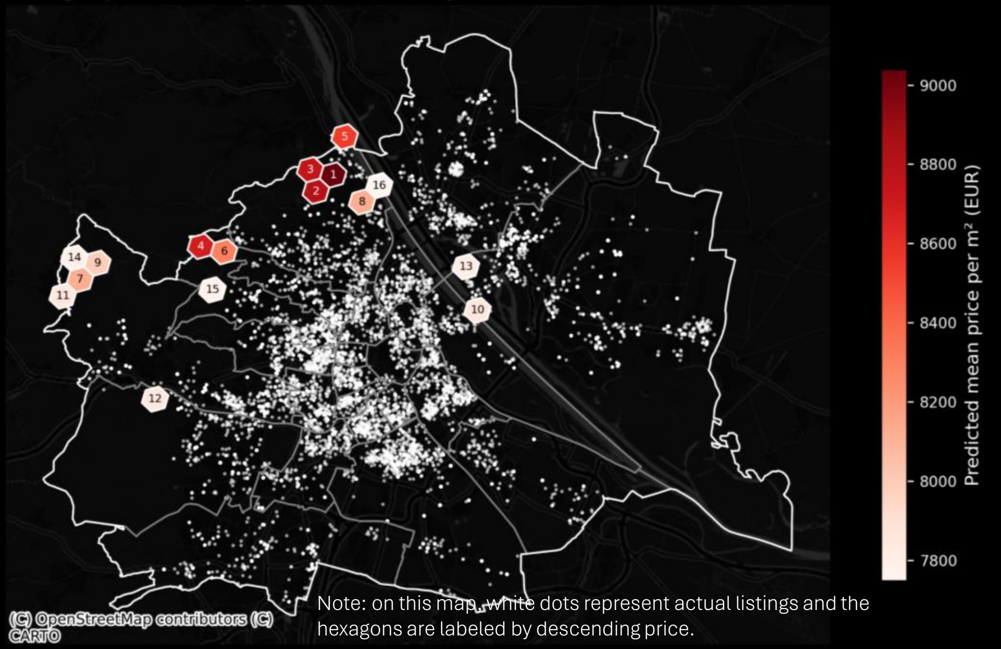
The most suitable areas were identified through a rules-based approach. I considered such areas suitable which had an average price larger than the 85th percentile (high potential), had less than the median number of listings in the scraped data (low competition), and were overlapping residential areas (legal possibility of development) – see these on the bottom map.

Predicted property prices in Vienna
(with random shuffle of properties)



16000
14000
12000
10000
8000
6000
4000
Predicted price per m² (EUR)
(C) OpenStreetMap contributors (C) CARTO

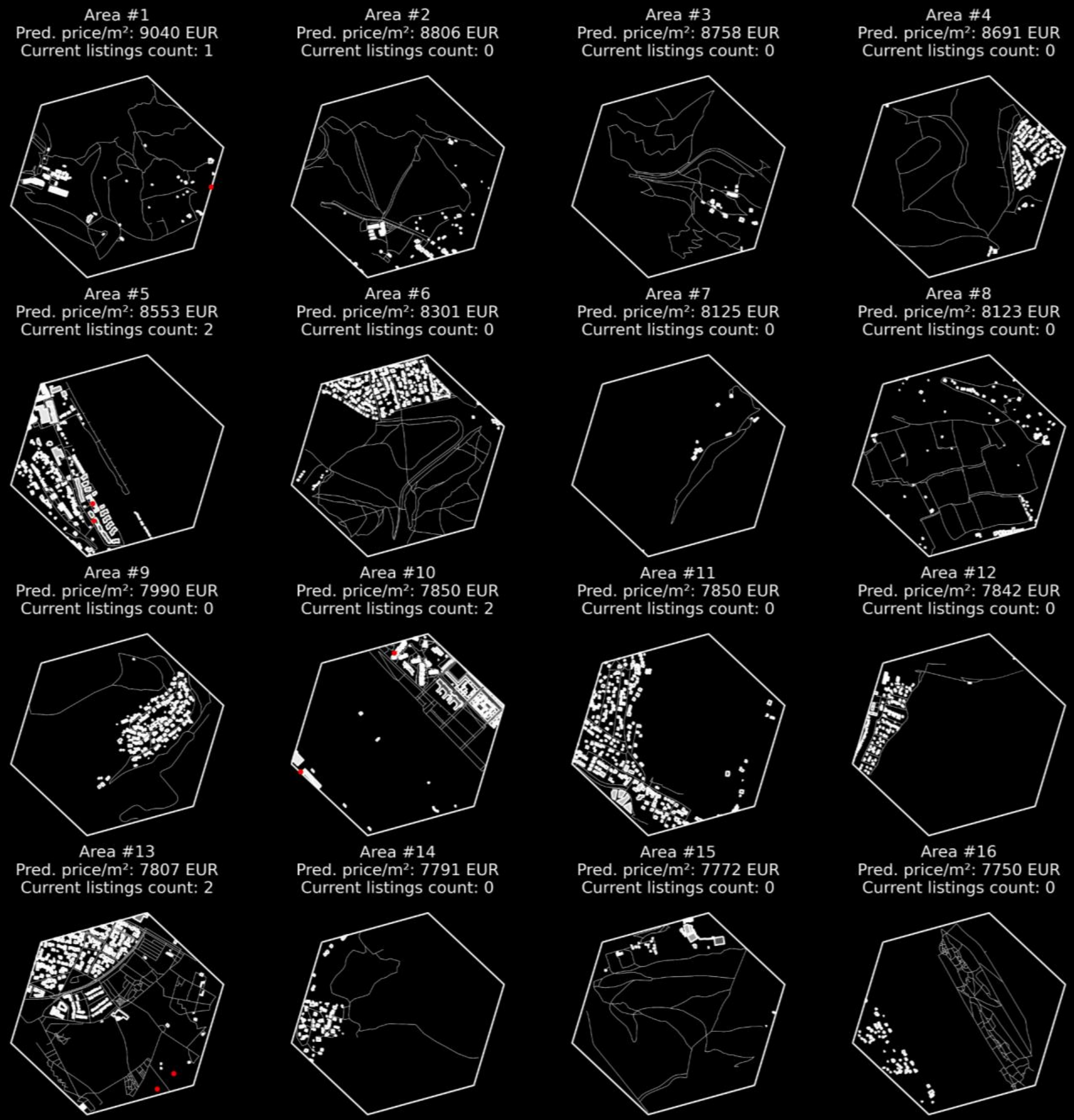
Most suitable areas in Vienna for real estate development:
high predicted price, low competition, residential areas



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
9000
8800
8600
8400
8200
8000
7800
Predicted mean price per m² (EUR)
(C) OpenStreetMap contributors (C) CARTO

Note: on this map, white dots represent actual listings and the hexagons are labeled by descending price.

Areas with highest predicted price/m^2 and low competition, overlapping residential areas
(current listings marked with red dots)



Area #1
Pred. price/m^2: 9040 EUR
Current listings count: 1

Area #2
Pred. price/m^2: 8806 EUR
Current listings count: 0

Area #3
Pred. price/m^2: 8758 EUR
Current listings count: 0

Area #4
Pred. price/m^2: 8691 EUR
Current listings count: 0

Area #5
Pred. price/m^2: 8553 EUR
Current listings count: 2

Area #6
Pred. price/m^2: 8301 EUR
Current listings count: 0

Area #7
Pred. price/m^2: 8125 EUR
Current listings count: 0

Area #8
Pred. price/m^2: 8123 EUR
Current listings count: 0

Area #9
Pred. price/m^2: 7990 EUR
Current listings count: 0

Area #10
Pred. price/m^2: 7850 EUR
Current listings count: 2

Area #11
Pred. price/m^2: 7850 EUR
Current listings count: 0

Area #12
Pred. price/m^2: 7842 EUR
Current listings count: 0

Area #13
Pred. price/m^2: 7807 EUR
Current listings count: 2

Area #14
Pred. price/m^2: 7791 EUR
Current listings count: 0

Area #15
Pred. price/m^2: 7772 EUR
Current listings count: 0

Area #16
Pred. price/m^2: 7750 EUR
Current listings count: 0

To make sure that the model makes sense, we should take a closer look at the predicted most suitable areas

A deep(er) dive into the identified areas

Visually, we can see from the previous slide's figure that the identified hexagons are clustered into 4 blocks (1-2-3-5-8-16, & 4-6-15, & 7-9-11-14, & 10-13) and one outlier (12). However, as most of the identified areas lie on the outskirts of the city (which is mostly a forest), we might want to check the local maps to see if the results make sense. Thus, I plotted building footprints are road networks in each hexagon, and indeed, it seems that real estate development would be possible in these areas.

Limitations

Though WillHaben is a very large data source, its coverage may be biased or distorted, which could distort the results. The trained price prediction model is pretty good, but it may have been further improved by more extensive tuning or trying out more algorithms. When interpreting predicted mean prices per area, one should keep in mind that this relates to the *average property in Vienna* which may or may not be relevant in certain areas (e.g. because the average property is on the 2nd floor). All in all, the results should be rather interpreted as recommended areas, which could then be further explored by experienced professionals.