Scraping process

In the following paragraphs our scraping efforts will be described. The scrapers can be examined in the attached Jupyter Notebooks labelled XX and XX.

Boliga

Our data comes from Boliga.dk, the largest independent online web-portal for real-estate sales in Denmark and has access to unique features such as "liggetid", price-development and access to BBR - the Danish Building and Housing Register (boligagruppen.dk 2019). Giving us unique insights into the pricing of real-estate in all 98 municipalities of Denmark. At the time of scraping 65,950 properties were for sale.

The scraping process was conducted on Friday the 23rd of August 2019. We had advised Boliga of our intent to collect data from their website via our scraper in an effort to identify ourselves and our intent (Shiab 2015). In order to scrape the data of interest we familiarized ourselves with the HTML-structure of Boliga. On the basis of these insights we constructed a code, which were able to scrape every page, containing information pertaining the currently listed real-estates on Boliga. The scraper requested all information available from each individual page, which surmounted to 65,950 URL requests.

For each URL, 34 features and the target variable (price) was collected. Table XX provides an overview of the features with a short description, and whether the feature has been dropped or saved for later usage. There are three reasons for a feature to be dropped:

1. The feature does not act with independent characteristics according to the our research,
2. The feature contains insufficient data,
3. The feature is poorly formatted and cannot efficiently be recreated.

Hvorlangterder.dk

The scraping of hvorlangterder.dk was achieved by writing a function that took in the GPS-coordinates gathered from Boliga and a list of points of interest. It then constructed an URL, which was then scraped and the json response was then returned as a dictionary of distances to the points of interest. The values from each key in the dictionary was then extracted as a new column in a Pandas DataFrame.

As Jupyter is bad for running long asynchronous tasks and an estimated running time of 18 hours this scraping procedure was run in Visual Studio,