

Aalborg University
Copenhagen



Faculty of Engineering and
Science

Development of a pricing tool for the real estate agencies in Copenhagen

Candidates: Wojciech Tomasz Bugajski
Thor Gabriel Svitzer Hansen

Supervisor: Frantisek Sudzina
Global Systems Design, Cand. Tech.

Number of pages: 72

June 2017

Academic Year 2016/2017

Contents

1. Abstract.....	4
2. Introduction	5
2.1. Objectives.....	6
2.2. Problem formulation	6
2.3. Structure.....	6
3. Research design	8
3.1. Limitations	9
3.2. Related work regarding the real estate market.....	10
4. Theoretical background.....	14
4.1. Sources	14
4.2. Greater Copenhagen housing market	15
4.3. Macroeconomic overview	17
4.4. Residential property market.....	17
4.5. Fundamentals of house prices	19
4.6. Section summary.....	21
4.7. Linear Regression Model.....	22
4.8. Data scraping theory	24
5. Methodology	30
5.1. CRISP-DM Model	30
5.2. Software	32
5.3. Data Scraping from Boliga.dk.....	34
5.3.1. Accessing the data.....	34
5.3.2. Explaining the code in Python.....	36
5.4. Preparing the dataset for a regression analysis	41
5.5. Data visualisation using Google Maps API	43
6. Analysis.....	48

6.1.	About Boliga.dk	48
6.2.	Flowchart – Overall process	49
6.3.	Data scraping results.....	50
6.4.	Visualization - Excel Power Map	50
6.4.1.	Preparing the data	50
6.5.	Data cleaning	54
6.6.	Variables in SPSS	55
6.7.	Predicting Values of Dependent Variables	57
6.7.1.	Dummy variables	60
6.7.2.	Model summary	60
	60
6.8.	Conclusion of SPSS analysis results.....	61
6.9.	Identifying overpriced apartments.....	62
6.10.	Plotting the apartments in Google Maps	64
6.11.	Section summary	67
7.	Conclusion	68
8.	Criticism & Future work.....	69
8.1.	Future work	69

Abbreviations

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BI&A	Business Intelligence and Analysis
CRISP-DM	Cross Industry Standard Process for Data Mining
CSS	Cascading Style Text
CSV	Comma Separated Values
DOM	Document Object Model
GDP	Gross Domestic Product
GNU GPL	GNU General Public License
GPS	Global Positioning System
HTML	Hypertext Mark-up Language
IDE	Integrated Development Environment
IP	Internet Protocol address
JS	Java Script
MRA	Multiple Regression Analysis
SPSS	Statistical Package for the Social Sciences
URL	Uniform Resource Locator

1. Abstract

This paper develops a method for identifying overpriced apartments based on historical data of sales in Greater Copenhagen Area. Data is obtained through web scraping a www.boliga.dk page and later analysed. The current real estate market in Greater Copenhagen is investigated and the fundamentals of the housing market are presented. Furthermore, the key market drivers are reviewed. A regression model is conducted and tested based on the data gained from the web scraping. The regression analysis has the purpose of identifying the overpriced apartments in Copenhagen. Evaluation of the results is executed and direction of further research is drawn. Our empirical findings confirm that our approach to the problem has been largely successful and is able to produce predictions that are of high value for real-estate agencies.



Figure 1 - Øresunds region.

2. Introduction

Copenhagen, the capital of Denmark, is located at the centre of the Oresund region and is one of the most densely populated, affluent and dynamic regions in northern Europe. According to Statistics Denmark, the capital region will experience a population growth of 18% over the next 25 years. This makes it the fastest growing region of Denmark by far. A population growth this significant creates a drive of growth in the housing market.

For many people, one of the most important decisions and purchases in their life is buying a property. This is why the trend in the housing market is often a topic of national attention. The housing market in many ways reflects the economic situation of a country, and a negative or positive trend can have a massive impact on the general population. When people consider buying a home, usually the location has been constrained to a specific area such as close to their workplace or to a school. Besides location, the housing prices are affected and restricted by the politics, economics, administration, nature and other factors.

Real estate is one of the most competitive markets in the world. Regardless of the prevailing economic condition – boom or bust – buyers are still looking for good investment opportunities and sellers are looking for ways to recoup their investment. Big data has also found its way into the real estate market. By analysing multiple data points, such as areas in the city and previous sales prices, it is possible to get a better read on the property as a whole and make accurate recommendations to clients.

In this thesis we present a modelling process for estimating and identifying properties that lie outside of the predicted price, using a multivariable linear regression model based on information gathered from web scraping the biggest real estate database in Denmark – Boliga.dk.

Many parties have a big interest in acquiring the most precise predictions of the market value of residential real estate. Real estate agencies benefit from targeting the housing market with the most accurate prices. It is in the interest of both buyers and sellers to set the prices as relatively to the market as possible.

2.1. Objectives

Purpose of the thesis is to create a prototype of a valuation tool for real-estate agencies in Copenhagen. The tool will provide the real estate agents with location-specific predictions and trends on the Greater Copenhagen real-estate market. It will contribute with the following characteristics:

- Improve real estate agents' decision-making through data analysis.
- Determine which parameters have the highest influence on the price-level of the apartments in Copenhagen.
- Allow real estate agents to estimate whether an apartment is overpriced.

2.2. Problem formulation

Purpose of this thesis is to answer the following question, which could potentially be of big interest for real-estate agencies:

- Is it possible to identify overpriced apartments through data analysis in the greater Copenhagen area?

2.3. Structure

The thesis structure is as follows:

Chapter 1 - covers the introduction and formal subjects such as objectives, problem formulation, structure of the thesis and limitations.

Chapter 2 - contains the theoretical background for this thesis, covering related work and the theory behind the previous and current state of the Greater Copenhagen real estate market. The fundamentals and key drivers for the real estate market are analysed and presented, to give the reader a profound and in-depth understanding that is required to comprehend the complexity of the real estate market. Furthermore, the theory behind web scraping is explained. Also, the theory behind linear regression is clarified briefly.

Chapter 3 - describes the methodology part. The Crisp-DM model illustrates how the paper is constructed and allows the reader to understand which approach is used. The regression analysis is conducted, the visualization part is both described and presented. Lastly, the software used for this purpose is explained.

Chapter 4 - is the analysis part where it is described how the data was grasped, cleaned up and used in SPSS for the analysis. A flowchart illustrates the whole process from beginning to the end results. The use of Google Maps API is also explained and the results are presented.

Chapter 5 - describes the overall conclusions of the thesis. Future work is taken into consideration and explained.

3. Research design

The science-theoretical approach in the dissertation will be described with help of "The research Onion" [1]. The model consists of six rings, which explains each part of the research process. The process starts in the outermost layer working its way in. The research onion contributes by adding a more structured method section.

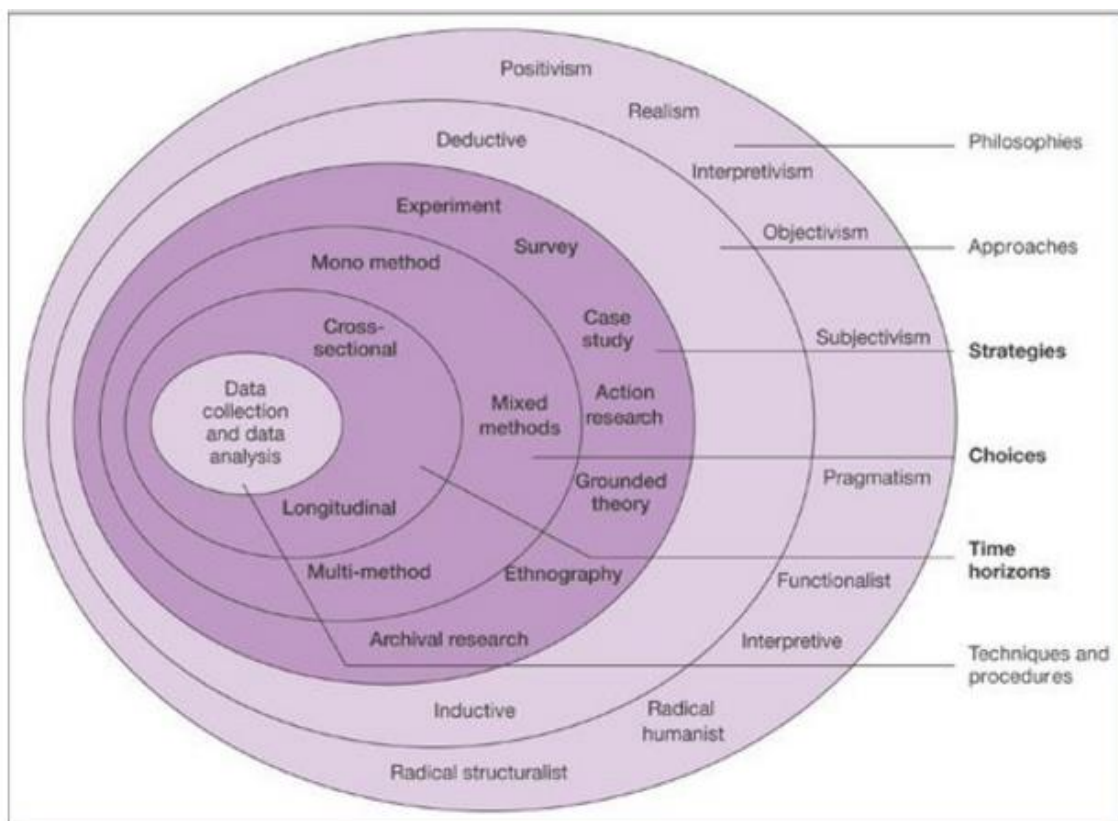


Figure 2 - Illustration of The Research Onion

The research onion

The outermost section describes how the investigators look at the world is stated. This paper is positivistic, meaning that the researcher's main focus is to observe and measure variables in certain controllable conditions and describe the results from these.

This paper will follow a deductive approach. When using deductive approach, you first find a possible connection between different variables, which in this paper will be done

based on existing theory [1]. Using quantitative data makes it possible to test different hypotheses.

This paper starts by being a case-study, investigating the related work that exists in this field and using the gained knowledge to create hypotheses. Different variables are selected and then tested through a regression analysis. Finally the results are upheld against the hypotheses to validate them. Additionally only quantitative methods are used with measurable and numeric data, with statistical comparisons.

The study is time-constrained from 2006-2017 thus the research is cross-sectional and therefore only serves as a snapshot of the situation. The final part of the research onion has to do with the data collection and analysis, which is explained in the section 5.3., page 32.

3.1. Limitations

In this section, we present the characteristics that influences the findings from this research. First of all, the data used for this thesis comes from one source that originally includes eleven variables (described in chapter 6.6.). Therefore, information about apartments like transportation feasibility, whether the building has been recently renovated and proximity to highly trafficked road is not taken into account regarding attractiveness of an apartment. However, the data obtained is still sufficient to identify the overpriced apartments. It should be considered in future work, what makes those apartments so attractive and try to find an answer by looking into additional features. In order to correctly constrain sample size, while keeping it sufficient, we decided to look into apartments sales in the Greater Copenhagen Area since 2006. The amount of data is big enough to find significant relationships. Data from boliga.dk also contains errors (described in 6.5.) that have to be manually inspected and deleted.

City transportation feasibility has been rejected as a possible variable that influences price sales, as it is impossible to trace back bus routes in the past and the way they have changed. When the metro construction is finished, in a near future, it will be possible to observe whether it affects the prices of the apartments in the area.

One of the limitations of this project is the finite amount of addresses that can be transformed into latitude and longitude by Google Maps. Maximum is 2500 requests per day [27]. Therefore, emphasis is put on visualizing data within one postal code. For

future use as a commercial application, it would be necessary to buy a subscription for Google Maps services.

3.2. Related work regarding the real estate market

A number of research papers were inspected in order to clarify the scope of the project. The literature about the Danish real estate market is not overwhelming. Furthermore, it is hard to find any specific literature about predicting housing prices. Therefore, it has been necessary to look elsewhere to find related scientific work. One of the papers reviewed is based on the real estate market in Stockholm while another one focuses on the real estate market in Los Angeles. While these two real estate markets are fundamentally different than in Copenhagen, many similarities can still be found. Furthermore, a deeper understanding of contemporary knowledge about the real estate market in Copenhagen is described in chapter 4.1., page 14.

The real estate market in Stockholm

Authors Robert Hu and Emil Sjögren wrote the paper 'Analysis and prediction of apartment prices in inner city Stockholm (2014)'. The paper is based on the real estate market of apartments in inner Stockholm. The main focus is on finding the best fitted analysis method, in order to predict the prices for an apartment based on its characteristics. Furthermore, the paper attempts to clarify which factors determine the value of the apartment. Data was acquired by contacting broker firms in central Stockholm. The data obtained consisted of 7000 sales, including different parameters such as:

- Initial price
- Price per m²
- Final price
- Amount of rooms
- Floor level

The data was based on sales from August 2012 to February 2014. Three models were compared:

- The Linear Model
- The Time Series
- The Combined Model

The conclusion was that both the linear model and the time series model are valid up to an accuracy of 66%. The linear model without outliers had a median accuracy of 90% and all the models were set around this level. They found that the most accurate model is the combined one without outliers. Besides that they found that it is more expensive to live in a central district of Stockholm and on a higher floor. Furthermore, they found some indications that autumn is a good season to sell and winter is optimal for buying.

The real estate market in Los Angeles

The paper 'Predicting the Market Value of Single-Family Residential Real Estate' (2015) by Roy E. Lowrance. The paper is based on the residential real estate prices in Los Angeles County in the time period 2003 through 2009. The main goal of the paper is to systematically design a local linear model that has the lowest expected error on unseen data. A wide range of linear models was used in order to find the most accurate. The parameters used in the linear model consisted of 24 different house and location features, such as:

- Living-Area
- Year-built
- Swimming pool
- Median-household-income
- Parking spaces
- Total rooms

From the 24 parameters 15 were found to be relevant.

The real estate market in London

The paper 'Machine Learning for a London Housing Price Prediction Mobile Application' written by Aaron Ng, presents a mobile application that can generate predictions for future housing prices in London. A number of regression methods was tested and compared in order to find the most accurate. The different models used includes:

- Bayesian Linear Regression
- Relevance Vector Machines (RVM)
- Gaussian Process (GP)

Of these the one that gave the most accurate results was the GP. By using GP as the choice of the prediction method, it was possible to produce predictive distributions instead of just point estimates. Besides from that, this paper included GPS coordinates in one of the models which provided a useful feature in predicting the real estate prices.

The real estate market in Istanbul

The paper 'Determining the Factors Affecting Housing Prices' written by Ezgi Candas, Seda Bagdatli Kalkan and Tahsin Yomralioglu tries to determine which parameters can be used as valuation factors. The city of Istanbul faces some obstacles that are not taken into consideration in the previous paper, such as:

- Population growth
- Illegally built houses
- Disaster-vulnerable buildings
- Infrastructure related problems
- Transportation issues

The study uses Multiple Regression Analysis to find which parameters are most significant. The test is performed 3 times, with the first test consisting of 19 parameters. Second test is conducted with the 15 most relevant parameters from the first test. Lastly the third test is performed with 5 parameters that are found most relevant. These parameters are:

- Floor
- HeatingSystem
- EartquakeZone
- RentalValue
- LandValue

The conclusion is that LandValue and RentalValue have the highest impact on the housing process. The other parameters are following them. The dataset used in this paper only consist of 116 data points. This sample size is very limited and it is concluded in the paper that it is very likely that the regression model will change drastically with a bigger sample size.

Quality of related work

Summary of the related work

The related work is relevant in many ways. The material has been analysed and the most useful statements have formed the basis for our own investigations. The main aspects of the related work are the following:

- The use of GPS coordinates to plot the real estate's instead of addresses
The use of geo-location is more useful than the actual address. With the use of Google Maps API it is possible to plot the exact location with the help of geo-location.
- Use long time periods of data sets
Our data stretches from 2006-2017. This amount of data gives us the necessary prerequisites to construct a valid analysis.
- Gives an insight on which linear models are useful/relevant
It is clear that the most useful method of analysing the selected variables are through a regression analysis.
- Gives an insight of which parameters are relevant to predict the housing prices
From the related work we have picked the most useful variables. These, together with the ones that are accessible from the database, forms the basis for our regression analysis. See chapter xx for a more profound explanation of the exact variables.
- Small dataset can be misleading

4. Theoretical background

The goal of this section is to give the reader a profound understanding of the Greater Copenhagen real estate market and the mechanisms that drives the prices. Furthermore, the theory behind the chosen regression method is explained and lastly, the theory behind data scraping is clarified.

4.1. Sources

This sections main property data sources, consist of the two leading property indexes in the Danish real estate market, and different scientific sources within the real estate market:

- Sadolin & Albæk
- Danmarks Statistik (DST)
- Scientific papers on the field

These sources are picked, since they give an objective and profound explanation of the key mechanisms that drives the real estate market in Copenhagen. It is estimated that these sources holds the highest representativeness because their analyses of the real estate market, are based on professional assessments and transparent transactions from the Copenhagen real estate market.

Sadolin & Albæk

The Sadolin & Albæk index is published by the Danish advisory and brokerage firm under the same name. First publication came in 1985, and the index covers different aspects of the Danish real estate market, with the prime focus on commercial and investment property' in the Capital region.

Danmarks statestik

DST also publishes a yearly index covering statistics on commercial and private real estate and has been doing so since 1992. The index is measured based on all real estate transactions registered in Denmark.

4.2. Greater Copenhagen housing market

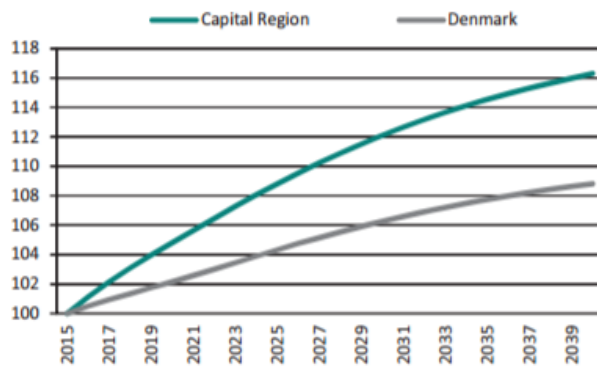
Danish economy is slowly on the way towards recovery. The unemployment rate is on a downwards stream and the consumer spending is going up. Nevertheless, the growth rate is still struggling and stays under the 2% mark, which indicates that the Danish economy still has a long way to go before it is up and running [2].

Since 2014 the prices have increases for houses in the Capital Region in the range of 5-6%. pr. Year [2]. The same development has occurred in the rest of the country, but at a lower level. However, it is in the market for condominiums, the big cities stand out in the form of high rises. Viewed over a longer period, price volatility in the capital has generally been higher than for the rest of the country as a whole. The amount of condominiums is, as in the rest of the country, at a very low level. The high turnover and low supply is helping to push up the prices in Copenhagen and pull the national average up. The past year's rising prices of condominiums in Copenhagen city has to be viewed in light of a growing population. Copenhagen and Frederiksberg's population have increased by 10-12.000 people a year since 2008. This is due to migration and -relocation, but also that have bigger amount of young people choose to stay in town, after they have started a family. In 2014, migration from the rest of the country was almost zero, while population growth was fortified by rising immigration and increased birth surplus [2].

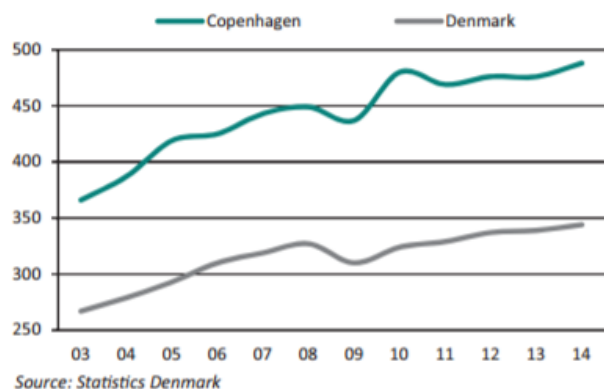
On the other hand, there is a number of newly built apartments in Copenhagen, which helps to increase the supply of condominiums, and thus puts a damper on price increases. Over the past four quarters, the amount of construction that were completed was nearly four times higher than the recent low point in the wake of the financial crisis in Q2 2010 [2].

Greater Copenhagen has a population of around 1.2 million. Combined with the Øresund region the count makes close to 3.8 million residents. One of the main drivers of growth in the housing and labor markets is population growth. According to Statistics Denmark the Capital Region of Denmark, is expected to grow 15% over the next 25 years [3].

Population forecast 2015-2040 (index 100 = 2015)



GDP per capita (DKK '000)



A new study from Eurostat shows that the Danish GDP per capita is more than 25% higher than the EU average. Furthermore, in Copenhagen the GDP per capita exceeds the national average by 25% [4]. Education-wise the residents of Copenhagen, that graduate with a university degree, exceeds the EU average with a whopping 74% above the average in EU capitals. The housing situation in Greater Copenhagen is still booming. A combination of strong demographics and low interest rates make the current housing market very attractive for both the public in general and also domestic and international institutional investors, property funds and companies. To sum-up some of the key drivers of the Greats Copenhagen housing market are [2]:

- Overall population growth fuels the boom in new construction, urban renewal and development of infrastructure.
- High concentration of high-income inhabitants and a strong business environment.
- Low level of interest rates. E.g. negative short-term interest rates or 10-year government property yields below 1,0% p.a.

4.3. Macroeconomic overview

Interest rates have domestically and internationally been falling for numerous years and in Denmark the long-term mortgage interest rate reached down to 2% in 2015. Since then the rate has moved up to 3%, but the current level still remains very low [5]. This applies even more so in the short end of the maturity spectrum. Low interest rates have through the recent past contributed to increasing activity in Denmark and stimulated housing prices. The further decline in the beginning of 2015 combined with an overall improvement in trading conditions lead to an acceleration in the house prices in the first months of 2015. This was the case for both single-family houses and condominiums. Developments in the annual rates of price increases shows, however, relatively large monthly fluctuations, so it may be difficult to derive the most recent tendency.

It is uncertain how much and how quickly progress in Copenhagen might spread to the surrounding areas. Such diversity will take the top of the price rises in Copenhagen when residents move away from the city. Historically, there has been a high correlation between trends in Copenhagen and the surrounding regions. It is not just a scattering effect, but is also due to the housing market in the country affected by the same underlying factors, including general economic development.

4.4. Residential property market

Only one in five of the Copenhagen residents own their home today. On national level 50% of the Danes prefer owning than renting a property. One major reason is that Copenhagen is a city with a significant amount of first time job hunters, students and senior citizens. All of them normally prefer flexible housing. Another major cause is to be found in the fact that a major share of the Copenhagen properties consists of relatively small units not suitable for families. Furthermore, the recent years urbanisation has changed the general migration pattern, where a larger amount of young families with small children (usually just one) is settling in the central city districts instead of the suburbs where most of them grew up [2].

One of the consequences of the economic crisis in 2007/08, was stricter requirements to obtain a loan of mortgage, for the Danish FSA demanded that homebuyers put up a down-payment of min. 5% of the full price. This fuelled the rental market in the following years. By 2014 this negative pattern was broken, by a combination of low interest and

growing confidence in an upcoming economic recovery. It triggered many households to swing back to ownership housing. This trend continued throughout 2015 and 2016 [2].

Ownership housing ratios

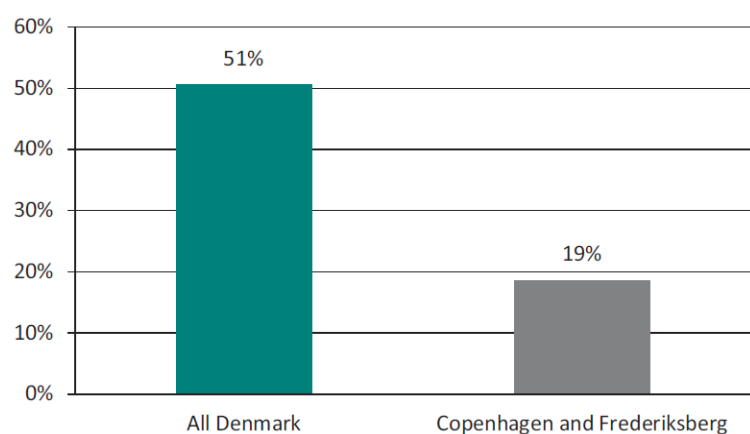
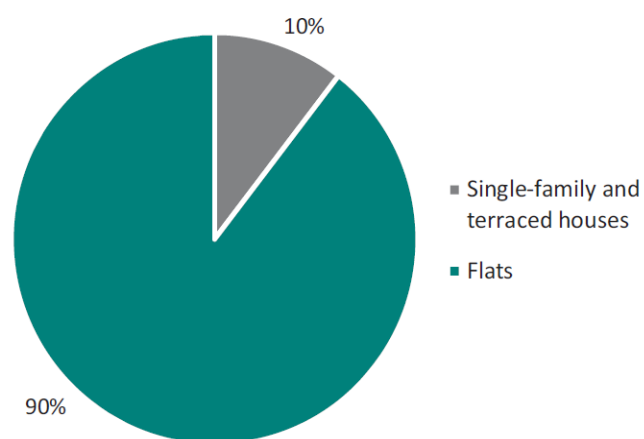


Figure 3: Housing stock in Copenhagen. Source: Statistik Danmark

Housing stock in Copenhagen and Frederiksberg, 2015



4.5. Fundamentals of house prices

Many factors determine the real estate prices. For example, an increase in the real estate market could be explained by a rise in the general disposable income. There is a lot of literature in this field and most of it suggest that the price of real estate is decided by short-run demand orientated variables and a long-run equilibrium [7]. Because of the volatile nature of house prices, most models incorporate a deviation from equilibrium that works as a corrective variable, this helps to focus on the permanent factors of the model.

The fundamentals of house prices consist of three main components. They consist of:

1. Real disposable income
2. House stock supply
3. Interest rate

Other important variables include, affordability, unemployment rate, credit supply and demographics. Another explanation that is widely recognized is house prices as a function of general income. This explanation is supported by investigating the co-integration between house prices and income, having income as the explanatory variable [8].

A study from 2010 - Blixen, Finecke, 2010, analyses which factors influence the most on housing prices and also accounts for how many models use the different factors. All the models used in the study, are from European countries, and therefore can be applied to Denmark. A table with the factors and their related elasticity towards the dependent variable are presented in the following page.

Elasticity of house-price towards factors	Elasticity	Estimated elasticity average	Nb. of Models using factor
Housing stock supply	-0.5 – 7.9	-3.2	11
Real disposable income	0.2 – 8.3	2.2	18
Demography	2.9	2.9	1
Real credit	0.2	0.2	2
Real wealth	0.4 – 0.7	0.6	2
Unemployment	-0.5 – 0.5	-0.07	3
Price Expectations	0.9	0.9	1
Construction Cost	1.8	1.8	1
CPI	0.8	0.8	1
Stock Index	-0.2 – 0.3	-0.13	2
Housing Affordability	0.3	0.3	1
Real interest rate	-1 – 9.42	-4.3	15

Figure 4:: Elasticity of house-price vs. factors

The table should be interpreted as follows: For example, the factor 'Real Disposable Income' is the most used variable (18 models uses this factor). On average it has a positive effect of 2.2% on the house price. This is perfectly logical thus wealthier population equals an increased demand of real estate services.

Another example could be the 'Housing stock supply' factor. This factor has an estimated elasticity average of -3.2 that should be read as the percentage change of the house price if a 1% change occurred to the factor.

The real interest rate is the second most analysed variable following real disposable income. The housing prices are affected negatively as a consequence of the interest rate. When the real interest rate raises, the loans get relatively more expensive, meaning a decrease in housing prices. Other important factors that influences the housing prices include demography, general unemployment rate and price expectation [9].

4.6. Section summary

The purpose of this section has been to examining and present, the factors and general key drivers that influence the Copenhagen real estate market. It was found that the real estate market is driven by the following factors:

- Overall population growth
- Limited house stock supply
- High concentration of high-income inhabitants
- Low level of interest rates

All of the above-mentioned factors will be used as part of the regression analysis, except of 'Limited house stock supply'. The reason behind this, is that the amount of data within the house stock supply field, is very limited. It is simply not possible to get hold of enough sources and data in this field, to obtain a valid conclusion.

Housing is an important commodity. The consequences of what happens in the housing market stretches far an inflicts not only the individual house owner, but the whole macroeconomic perspective. The most important aspects include:

- Makes up an important part of the Gross Domestic Product (GDP).
- The stock of housing and building constitutes as a great part of the overall national wealth.
- Changes in the housing market effects not only the value but also the distribution and composition of national wealth [6].

4.7. Linear Regression Model

The linear regression model represents the relationship between two or more variables, which are described by this formula:

$$X\beta + e = y$$

In this formula y is the regressand, which is known as the response variable or the dependent variable. The X represents the covariates, or the independent variable. If there is only one covariate the regression will be known as “simple”, if there is many it is called “multiple linear regression”.

The β 's are the coefficients and are the ones we are looking for. Using previous observed data they can be estimated and once found they represent the relationship between y and X 's. Lastly e represents the error term which captures factors that influence y but are not included in the model [10].

Multiple Regression Analysis

Multiple regression analysis (MRA) is a technique that allows multiple factors to be calculated in the analysis separately. By doing so the effect of each factor can be estimated. This is valuable information in order to quantify the impact of numerous influences upon a single dependent variable.

MRA is a highly used method within data analysis, to examine the relationship between the dependent variable and the independent variables. A multiple regression equation to predict “ y ” can be expressed as follows [10]:

$$y = \beta_0 + \beta_1x_1 + \dots + \beta_nx_n + \varepsilon$$

Dummy variables

Often a covariate is not a quantity, but an attribute. In order to see the effects of the attribute it is necessary to implement a technique called dummy variable.

X will be a covariate that is an attribute. In case the elements in x are active they would be represented by 1, if there are no effect it will be represented by 0.

In this paper, the dummy variable was tested to distinguish the apartments where the price was set above the predicted sales price. The results are presented in chapter 6.5.1

4.8. Data scraping theory

Screen scraping is a technique by which a computer program extracts the data from another program's output. The program used for this purpose is the so-called screen scraper. The main distinguishing feature of screen scraping from parsing is that the output of the scraping program is intended for human, and not for machine-based interpretation. Ideally, a replica of table from boliga.dk would be presentable in Excel file. Human-readable format is often encoded as ASCII or Unicode text. There are many synonyms of screen scraping: data scraping, data extraction, web scraping, page scraping, and HTML scraping (the last three refer to web pages).

Typically, data transfer between programs takes place through data structures adapted to machines rather than humans. Such structures and protocols are usually well-ordered, easily parsed, and limit ambiguity and duplication to a minimum. Very often they are unreadable to a man. Human output is the opposite of the above - formatting, redundant labels, comments and other information are not just superfluous, but can also interfere with data interpretation. However, if the output is only available in such a user-friendly format, screen scraping becomes the only automated way to transfer data. This term originally referred to the reading of data from the terminal screen memory. Similarly, screen scraping also means computerized HTML processing on web pages. In any case, screen scraper must be programmed not only to process interesting data, but also to reject unwanted information and formatting.

Web Scraping

In case of extracting data from content of HTML webpage, stored in text, HTML scraping is a relevant option. Data is stored in an encoded text form (ASCII or Unicode), designed for human end-users, not for automated use. Purpose of this HTML scraping is to download the data from boliga.dk and save it into CSV file which could be later transformed into excel file, that allows to run Regression Analysis and provides necessary data to visualize it and answer research questions. Large websites usually use defensive algorithms to protect their data from web scraping and limit the number of requests an IP may send [11]. Websites like boliga.dk have large amounts of pages that are dynamically generated from an underlying structured source like a database. For example, looking up books in Amazon clearly shows that it presents author, title and comments in the same way in all its book indexes. That kind of data is typically encoded by a common script or template. In recent years, the websites like comparison

shopping have emerged from extraction technologies. They create value-added services for private as well as business users.

In the literature review, we focus on topics that regard data scraping accuracy, choosing the best software/programming language and legality of data scraping from data hosts.

Arasu and Molina [12] present a great framework on how to inspect webpages in order to find the relevant and valuable information hidden in HTML web pages. They describe how the data is transferred from an underlying structured source like database to HTML file. The paper studies the problem of automatically extracting structured data encoded in a given collection of pages. Authors describe challenges in deducing the template such as differentiating between text that is part of the data and text that is part of the template. An algorithm for tackling the problem is presented with an extensive description of how it works. Authors also show schema for page creation. It is required for a web scraper to understand what does HTML markup consists of – tags, character-based data types and entity references. Some of the tags such as line break `
` do not permit any embedded content. Understanding the HTML architecture is necessary to scrape relevant data. Authors present the results of their algorithm – „on average around 80% of the attributes were extracted correctly.”

Another research, conducted by Hirschey [13], focuses on legality and acceptance of data scraping. He describes that the trends in data scraping seem to be reversing, as in the past scrapers were reposting information in order to compete with the scraped website and currently scrapers can offer mutual benefit which has the potential to enhance service provided by the scraped website. Increasingly symbiotic relationship between data scrapers and data hosts only adds to difficulty to clearly establish legal background of scraping case law. Author describes technical background on scraping, analyses legal protection for databases, drafts the key legal claims against scrapers and examines scraping litigations in order to determine when the suits have been successful/unsuccessful for data hosts. He focuses his work on US and international law.

Among the web scraping services that have mutual benefits for both the scraper and data host are web search engines. They pull data of search terms user enter to link the user to relevant webpage results. Even though they are web scraping service, they

have avoided the blame associated with scraping because they are "an instrumental part of the online ecosystem". In example Google's PageRank is thought to be the biggest scraping system that scrapes data from billions of webpages [14]. The problem that webmasters often face is the high consumption of Googlebot transfer. It may cause the pages to use their transfer limit and will be suspended for some time. It is a problem especially with mirroring pages that store gigabytes of data. Google allows access to "Webmaster Tools" which allow site owners to adjust the "intensity" of Googlebot's visit on the web page [15] Another valuable information provided by Hirschey regards technical background of scraping. Because it extracts data from HTML template (code is written accordingly to a scraped webpage), it is unstable to changes by data hosts.

In regards to the legality of web scraping, author presents information that airline price aggregators such as Kayak, Orbitz and Expedia have all been subject to legal action [16]. Facebook has history of suing third-party applications that have accessed and republished its' users' data [17]. It is interesting to see that Craigslist which alleged services like Padmapper, 3Taps of improper gathering of their information and reposting it as a map interface that plots the location of the user-generated ads, has evolved their business with a similar mapping service. Author states there is "no direct legal protection for databases" [18] However, data hosts can sue scraper if they can prove the scraper has harmed them. Case *Intel vs. Hamidi* is a precedent that ruled that server inconveniences do not constitute an actionable harm [19]. Scraping may slow the processing power of websites and in extreme cases crash a website or server.

To conclude, Hirschey says that multiple instances of data hosts pairing up with scraper show that data host should seek ways to embrace scrapers that seek to improve their services. He also notes that scrapers should review their business model. If a data host thinks scraper is parasitic, then he can sue the scraper.

Chen, Chiang, Storey [20] chime in with a thorough research regarding Business intelligence and analytics (BI&A). It has become increasingly important topic in the business communities. Big Data Analytics is identified as one of the four major technology in the 2010s. Survey ran by Bloomberg Businessweek (2011) depicts 97 percent of companies with revenues exceeding \$100 million do their business using business analytics. It allows companies to better understand their business and market. With the better understanding of goals and limitations, managers can make

timely business decisions. Authors describe characteristics and capabilities of BI&A, and companies shift into data-oriented business with well-structured data stored in commercial relational database management systems. Growth of the Internet has stimulated e-commerce businesses such as eBay or Amazon. Actions such as product placement optimization, customer transaction recommendations or customer relationship management can be currently accomplished through web analytics. A unique opportunity for companies to treat the market as a “conversation” between businesses and customers instead of one-way “marketing” is believed to be accomplishable due to social media analytics.

Web and e-commerce communities have created the buzz surrounding BI&A and Big Data. They have also led to significant market transformation with innovative e-commerce platforms and product recommender systems (eBay, Amazon). Internet firms such as Google or Facebook continue to drive the development of web analytics and cloud computing. Data Analytics have also had a wide influence on other fields such as: Smart Health and Wellbeing, Security and Public Safety (criminal network analysis), Science and Technology (mathematical and analytical models). As one of the foundational technologies in Web Analytics author presents web crawling, web services and recommender systems – all of which consist of data scraping.

Author presents graph which shows increasing meaning of both Business Analytics and Big Data in publications from 2000 to 2011.

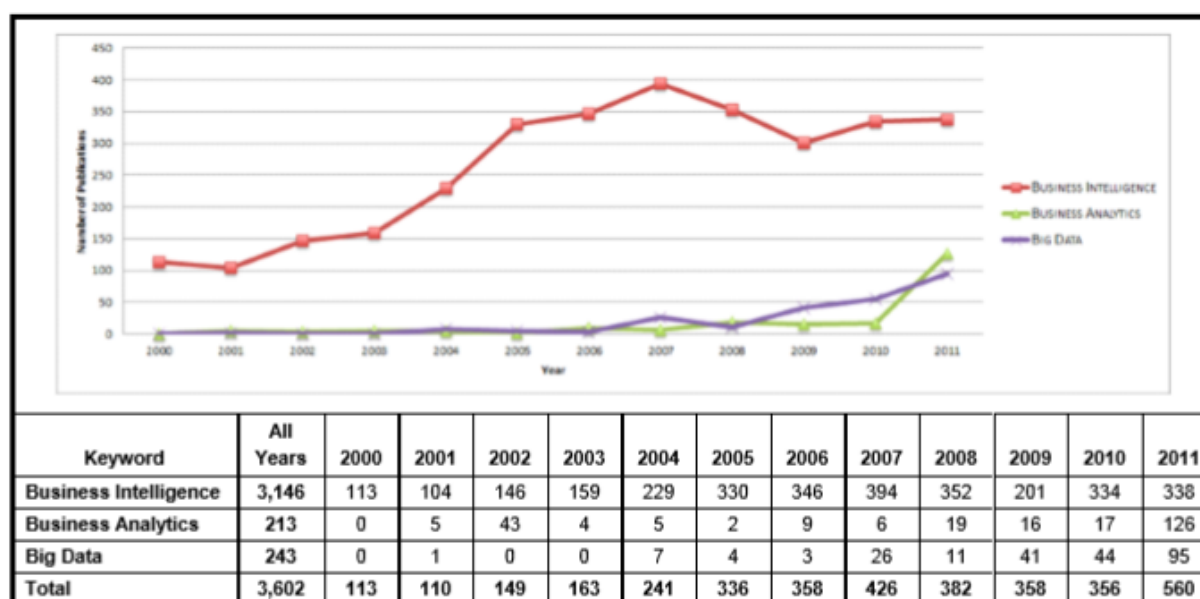


Figure 5: BI&A related publication trend from 2000 to 2011. Source [20]

The last paper [21] provides overview to web scraping technologies. Author reviews scraping frameworks and tools while identifying their effectiveness and weaknesses in terms of extraction capabilities. Glez-Pena gives overview of what does web scraping consist of:

- Site access through HTML protocol
- Html parsing and contents extraction
- Output building

That information is essential to create a working web scraping software. Author proceeds to describe common libraries (Scrappy, BeautifulSoup, Apache HttpClient) and frameworks for writing a web scraper. Author also says that although web scraping has lost some of the need for it, due to proliferation of web services, there is still a number of scenarios where it is the only available method to get the data from the host (i.e. whenever API is not available or API do not give access to the desired data). What is more, author presents the division of web scraping approaches:

- Libraries for general-purpose programming languages
- Frameworks
- Desktop-based environments

Information found in table below allow to see strengths and weaknesses in open-source web scraping libraries.

	Type C: HTTP client P: Parsing F: Framework	Domain-specific language	API/stand alone	Language	Extraction facilities R: Regular expressions H: HTML parsed tree X: XPath C: CSS selectors
UNIX shell (curl/wget, grep, sed, cut, paste, awk)	CP	No	SA	bash	R
Curl/libcurl	C	No	Both	C++ bindings	
Web-Harvest	F	Yes	Both	java	RX
Jsoup	CP	No	API	java	HC
HttpClient	C	No	API	java	
jARVEST	F	Yes	Both	Java/Ruby	RXC
WWW-Mechanize	CP	No	API	Perl	RX
Scrapy	F	No	Both	Python	RX
BeautifulSoup	P	No	No	Python	H

We have selected several available Web scraping packages oriented to programmers. There are six libraries implementing an HTTP client (C) and/or HTML parsing/extraction (P) and three frameworks (F). Web-Harvest and jARVEST frameworks present a domain-specific language for defining robots, based on XML and Ruby, respectively. For all the analyzed alternatives, we report their extraction facilities, including regular expressions (R), HTML parsed tree (H), XPath expressions (X) and CSS Selectors (C).

Figure 6 – Open-source web scraping libraries and framework. Source [21]

The literature review gives valuable inputs in regards to choosing the proper software for performing data scraping. It allows to choose between general-purpose libraries or a more integrative solution which is a framework. Comparison gives a better perspective of what library suits the best for the purpose of this thesis. Another important aspect discovered from literature is understanding how is data scraping perceived in the letter of law. It is clear to see, that violation of data hosts' rules can lead to filing a suit against scraper as it has happened in situations described in this chapter.

5. Methodology

The method section aims to provide a structured overview of the studies and the method used to answer the problem formulation. This section gives the reader an understanding of how the thesis has been conducted and how the empirical data of the thesis is collected. At the same time, this section gives the reader the ability to reflect on the choice of methodology, including the consequences of the methodological method.

5.1. CRISP-DM Model

Cross Industry Standard Process for Data Mining, also known as CRISP-DM is a model used for describing the most common approach that experts use for tackling problems. It provides a structured approach to planning data mining projects. The model consists of 6 steps: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation & Deployment. [22]

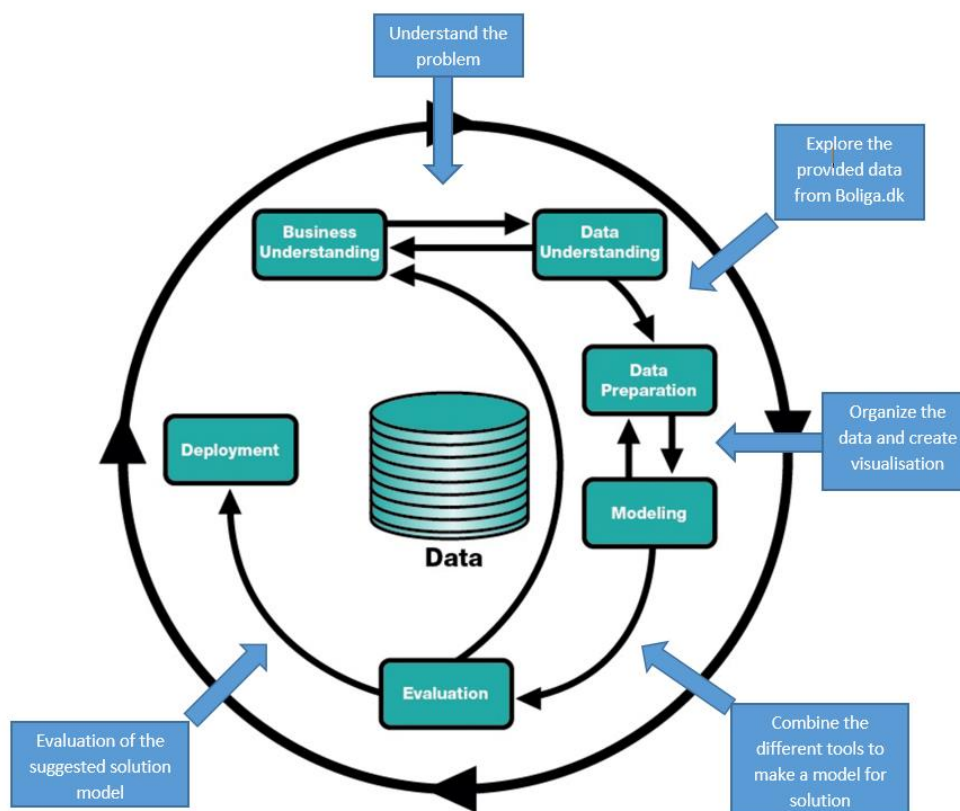


Figure 7 - Crisp-DM model.

Business Understanding

First step is to get a basic understanding of the business that one wants to examine. This can be done by methodically reviewing the following stages:

In the first step, we examine how the current real estate market is in Denmark and the greater Copenhagen area. It is important to understand the basic mechanisms of the housing-market in order to perform the correct analysis.

Data understanding

After gaining a basic understanding of the company the following step is to collect the available data and create the overview of this. In order to understand the data, we looked into Boliga.dk database. We looked into which variables were available and used the ones we found most relevant to our project. Boliga.dk does not provide any API, which means that the data can only be obtained by scraping the webpage.

Data preparation

The very first step is to collect all the data from Boliga's webpage. This was done using scraping code written in Python language.

Next, we save it in Excel and search for errors in the output. The data is dissected in order to look for irregularities and deviations due to errors in the database. These errors are then corrected in Python.

Modelling - SPSS analysis

In this step, the analysis is run in SPSS based on the acquired data from the previous steps. In the SPSS analysis outliers (errors found by authors) are found and removed from the dataset. Next, the overpriced apartments are identified.

Output visualisation using Google Maps API

The output from the SPSS analysis is then presented in a heat-map using Google Maps API.

Evaluation

The model is checked for validation in this step. We check whether we achieved the required goals and whether the model performs in the desired way.

Deployment

Further development is discussed in this step. Questions are raised on how the model can be improved so it adds more value to real estate agencies' business.

5.2. Software

Jetbrains Pycharm software

PyCharm - integrated development environment (IDE) for the Python programming language, developed by JetBrains company. It provides: editing and analysis of source code, graphical debugger, launching unit tests. It also supports programming and creating web applications in Django, as well as live editing of HTML, CSS and JavaScript (only in professional edition). It can be downloaded at:

<https://www.jetbrains.com/pycharm/download/#section=windows>

Python is a high-level programming language with extensive libraries to choose from [23]. Python implements several paradigms simultaneously. Similarly to C++, Python does not enforce a single programming style, allowing different uses: object-oriented programming, structured programming and functional programming are possible.

Types are checked dynamically, and garbage collection is used for memory management. Although its popularity is based on the differences with Perl, Python is in many ways similar to it. However, the designers of Python rejected the complex Perl syntax for a more cost-effective and, in their opinion, more readable syntax. Although similar to Perl, Python is sometimes classified as a scripting language, and is used to create large projects such as Zope Application Server or the Odoo (Enterprise Resource Planning / Customer Relations Management) software.

Python was designed to provide as much readability as possible in the source code. It has a simple text layout, uses indentation or English words where other languages use punctuation and has far less syntactic constructions than many structural languages such as C, Perl, or Pascal.

BeautifulSoup is a HTML and XML parsing library which also allows user to fetch contents from URL and to parse certain parts of them. As of 05.05.2017 fourth version is available (BeautifulSoup 4.5.3).

It has a great use for obtaining data through web scraping. It only fetches the contents of the URL that user gives and then stops. It does not crawl unless it is manually put it inside an infinite loop with certain criteria. It allows to choose certain tags from HTML file from which user wants to extract data.

Installing BeautifulSoup:

BeautifulSoup4.5.3 is compatible with Python 3.6. It is an included package that is installed with command:

```
>>pip install beautifulsoup4
```

SPSS

SPSS is the acronym of Statistical Package for the Social Science. SPSS is a software used worldwide within the statistical analysis of data. SPSS was originally developed by SPSS Inc. and acquired by IBM in 2009. In addition to scientific research it is often used in market and opinion research as well as epidemiological studies.

SPSS is a great tool to determine the factors that affect the value of real estate. Some of the factors found through the SPSS analysis will only have a minor significance, while others will influence greatly on the value of the real estate.

Notepad ++

Notepad ++ is an extensive text editor based on the Scintilla project, distributed under the GNU GPL license. It highlights syntax of VB / VBScript, Unix shell scripts, BAT, SQL, Objective-C, CSS, HTML, Pascal Perl, Python, Lua, Ruby, Lisp, Scheme, Diff, Smalltalk, PostScript and others. In addition, the editor user can create colouring for his or her programming language using the built-in system.

The program supports autocomplete, search and replace strings with regular expressions, split screen editing, zooming, tabs (the ability to open several files at the same time). The program also allows you to create macros and plugins. A useful option is to search and highlight pairs of parentheses (opening and closing).

5.3. Data Scraping from Boliga.dk

In order to conduct the thesis, data regarding sales of apartments in Copenhagen was necessary to be obtained. Boliga.dk is a freely accessible database of every apartment or house sold in Denmark since 1992. Data can be sorted by post number, street, type of accommodation and year of sale. Since the webpage does not provide an API, the data needs to be scraped.

Adresse / Postnr	Købesum	Dato / Type	kr/m²	Rum	Boligtype	m²	Bygget	%
Bentzonsvej 41, ST. TH 2000 Frederiksberg	1.800.000	23-03-2017 Fam. Salg	22.500	3	Lejlighed	80	1900	
Bispeengen 13, 2. TH 2000 Frederiksberg	1.105.000	23-03-2017 Fam. Salg	16.250	2	Lejlighed	68	1911	
Stockflethsvej 39, 2. TV 2000 Frederiksberg	1.700.000	21-03-2017 Fam. Salg	26.984	1	Lejlighed	63	1934	
Kong Georgs Vej 51, 5. TV 2000 Frederiksberg	3.652.319	20-03-2017 Alm. Salg	44.540	3	Lejlighed	82	1963	-2 %
Folkvarsvej 7, 2. TH 2000 Frederiksberg	1.600.000	17-03-2017 Alm. Salg	25.396	3	Lejlighed	63	1888	-36 %
Nimbusparken 7, 2. 2 2000 Frederiksberg	4.695.000	17-03-2017 Alm. Salg	43.073	4	Lejlighed	109	2005	
Porcelænshaven 5H, 3. TH 2000 Frederiksberg	8.985.000	16-03-2017 Alm. Salg	65.583	4	Lejlighed	137	2006	0 %
Langelandsvej 51, 3. TV 2000 Frederiksberg	1.920.000	13-03-2017 Fam. Salg	29.538	3	Lejlighed	65	1897	
Holger Danskes Vej 16, 2. TH 2000 Frederiksberg	935.000	09-03-2017 Fam. Salg	19.893	2	Lejlighed	47	1885	
Fuglebakkevej 77 2000 Frederiksberg	12.400.000	08-03-2017 Alm. Salg	39.743	9	Villa	312	1914	-4 %

Table 1 - Boliga.dk database

5.3.1. Accessing the data

Combination of fn+f12 in Google Chrome browser opens the Developer Console. It can also be accessed by hovering over an item on a webpage and selecting „Inspect Element” from context menu. Google Chrome, as other web browsers, has support for web developer tools, which allows web designers and developers to look at the make-up of their pages. It is a built-in tool into the browser which does not require additional setup and configuration. HTML and DOM viewer is included in the built-in Development Tools. HTML and DOM viewer allows to see the DOM as rendered in addition to see which classes, JavaScripts (JS) are parts of the webpage. Apart from selecting and editing, the HTML elements panels demonstrate properties of DOM object, such as dimensions and CSS properties. Which is a necessary part of inspecting a webpage as it typically loads additional content in forms of URL links, images, scripts, fonts.

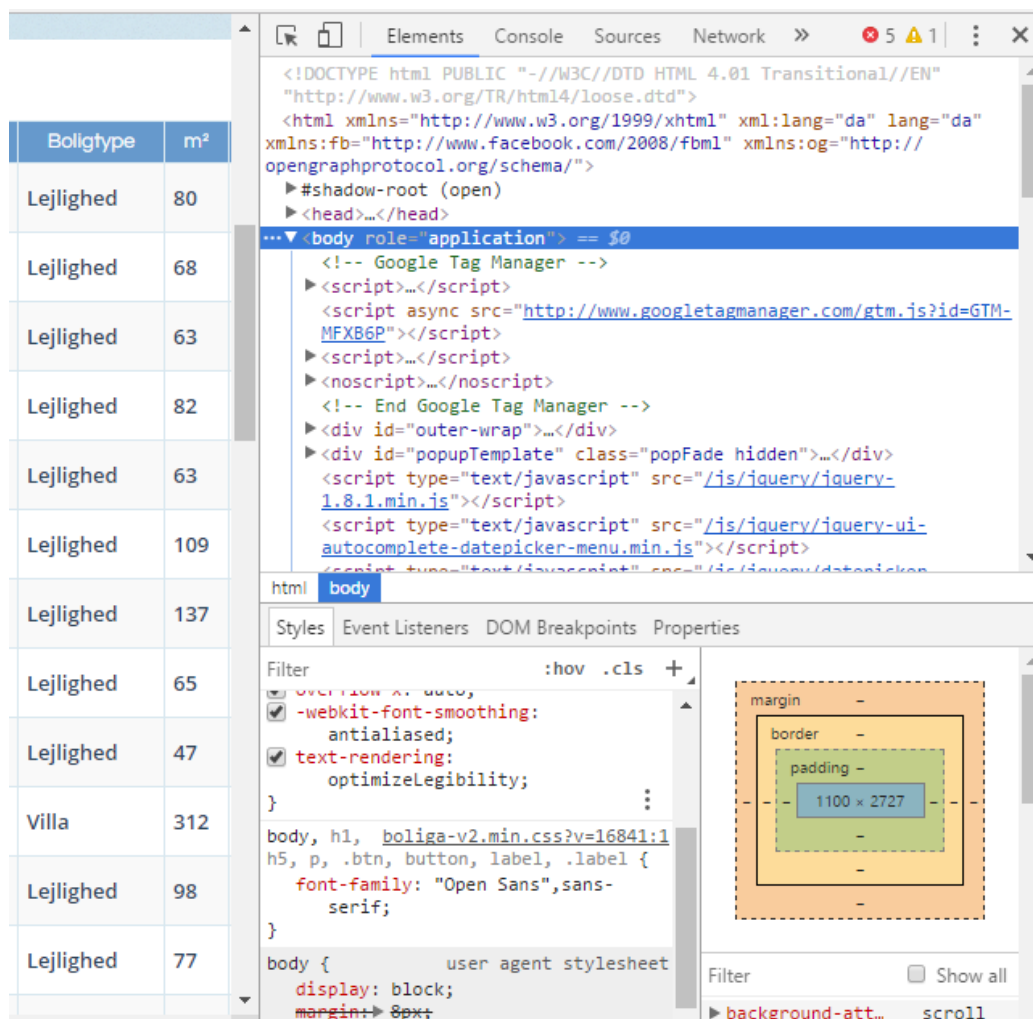


Figure 8 – Developer Console in Google Chrome.

The Elements panel depicted below allows us to see which part of database is written in which element of HTML. It needs to be identified in order to scrape the required information from table to CSV file.

4.288 resultater < 1 2 3 4 5 6 >

Adresse / Postnr	Kebesum	Dato / Type	kr/m²	Rum	Boligtype	m²
Bentzonsvej 41, ST, TH 2000 Frederiksberg	1.800.000	23-03-2017 Fam. Salg	22.500	3	Lejlighed	80
Bispeengen 13, 2. TH 2000 Frederiksberg	1.105.000	23-03-2017 Fam. Salg	16.250	2	Lejlighed	68
Stockflethsvej 39, 2. TV 2000 Frederiksberg	1.700.000	21-03-2017 Fam. Salg	26.984	1	Lejlighed	63
Kong Georgs Vej 51, 5. TV 2000 Frederiksberg	3.652.319	20-03-2017 Alm. Salg	44.540	3	Lejlighed	82
Folkvarsvvej 7, 2. TH 2000 Frederiksberg	1.600.000	17-03-2017 Alm. Salg	25.396	3	Lejlighed	63
Nimbusparken 7, 2. 2 2000 Frederiksberg	4.695.000	17-03-2017 Alm. Salg	43.073	4	Lejlighed	109
Porcelænshaven 5H, 3. TH 2000 Frederiksberg	8.985.000	16-03-2017 Alm. Salg	65.583	4	Lejlighed	137
Langelandsvej 51, 3. TV 2000 Frederiksberg	1.920.000	13-03-2017 Fam. Salg	29.538	3	Lejlighed	65
Holger Danskes Vej 16, 2. TH 2000 Frederiksberg	935.000	09-03-2017 Fam. Salg	19.893	2	Lejlighed	47
Fuglebakkevej 77 2000 Frederiksberg	12.400.000	08-03-2017 Alm. Salg	39.743	9	Villa	312

The developer tools on the right show the DOM structure, highlighting the table element with id="searchresult" and class="searchResultTable".

Figure 9 – Identifying relevant elements.

4.288 resultater < 1 2 3 4 5 6 >

Adresse / Postnr	Kebesum	Dato / Type	kr/m²	Rum	Boligtype	m²
Bentzonsvej 41, ST, TH 2000 Frederiksberg	1.800.000	23-03-2017 Fam. Salg	22.500	3	Lejlighed	80
Bispeengen 13, 2. TH 2000 Frederiksberg	1.105.000	23-03-2017 Fam. Salg	16.250	2	Lejlighed	68
Stockflethsvej 39, 2. TV 2000 Frederiksberg	1.700.000	21-03-2017 Fam. Salg	26.984	1	Lejlighed	63
Kong Georgs Vej 51, 5. TV 2000 Frederiksberg	3.652.319	20-03-2017 Alm. Salg	44.540	3	Lejlighed	82
Folkvarsvvej 7, 2. TH 2000 Frederiksberg	1.600.000	17-03-2017 Alm. Salg	25.396	3	Lejlighed	63
Nimbusparken 7, 2. 2 2000 Frederiksberg	4.695.000	17-03-2017 Alm. Salg	43.073	4	Lejlighed	109
Porcelænshaven 5H, 3. TH 2000 Frederiksberg	8.985.000	16-03-2017 Alm. Salg	65.583	4	Lejlighed	137
Langelandsvej 51, 3. TV 2000 Frederiksberg	1.920.000	13-03-2017 Fam. Salg	29.538	3	Lejlighed	65
Holger Danskes Vej 16, 2. TH 2000 Frederiksberg	935.000	09-03-2017 Fam. Salg	19.893	2	Lejlighed	47

The developer tools on the right show the DOM structure, highlighting the specific cell in the table with the value '26.984'.

Figure 10 – Identifying cells which include data.

5.3.2. Explaining the code in Python

This part of code is requesting library necessary for opening URLs as well as BeautifulSoup library for parsing the text from the webpage.

```
from urllib.request import urlopen as uReq
from bs4 import BeautifulSoup as soup
```

Following is the part responsible of creating a csv file in project folder. Data will be saved into this file.

```
filename = "apartmentss2930.csv"
f = open(filename, 'a') # a stands for append
```

Then a loop is opened. At each iteration, a maximum of 60 pages is scraped because if the number is exceeded, then the connection to the website is lost. Following error appears:

TimeoutError: [WinError 10060] A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond

Error happens because boliga.dk has a software which disables connection to programs that download the data from it. In order to tackle the following problem, function `sleep()` is included in the loop with every 60th page. It suspends execution of the program for the given number of seconds. From experiments with data scraping using the code, it has been observed, that with sleep time of 300 seconds program does not lose connection.

For loop is opened and its variable is transferred to the URL, so with each iteration of the for loop, the next page with data is scraped and saved into CSV file. Fortunately, page numbers with sales data only differ by one digit at the end of address. Connection to the webpage is established. Using the BeautifulSoup function the HTML page is parsed and all the data inside the cells of the table is read.

```
for num_of_pages in range(1, 180):
    # loop variable transferred to url
    my_url =
    'http://www.boliga.dk/salg/resultater?so=1&sort=omregnings_dato-
    d&maxsaledate=today&iPostnr=2300&gade=&type=&minsaledate=2010&p={}'.
    format(num_of_pages)
    # opening connection, grabbing the page
    uClient = uReq(my_url)
    page_html = uClient.read()
    uClient.close()

    # parsing html
    page_soup = soup(page_html, "html.parser")
    # Find tables in html
    tables = page_soup.findAll("table", {"class":
    "searchResultTable"})
    rows = page_soup.findAll("tr")
    print (len (rows))
    cells = page_soup.findAll("td")
```

```

print (len(cells))

if num_of_pages == 60 or num_of_pages == 120 or num_of_pages == 180
or num_of_pages == 240:
    time.sleep(300)

    #main loop
    x=0
    while x <40:
        .
        .
        .
        MAIN FUNCTION HERE

f.close()

```

Inside the For Loop there is a While Loop, which contains the main function. It reads all the data inside the table from boliga.dk and saves it into values separated by comma, which later allows to transfer it easily into Excel file. Each of the variables takes out the data from cells, the name is according to what is inside the cells in original table. Original table has 10 columns, therefore 'x' variable is transferred into output cells as a multiplicity of itself. That allows to cover all of the cells from the original table.

```

while x <40:

    cells = page_soup.findAll("td")

    adress_apt = cells[10 * x].get_text(separator="
").replace("ö", "o").replace("é", "e").replace(",", "
").replace("Æ", "0m0").replace("Å", "0n0").replace("Ø",
"0b0").replace("æ", "0v0").replace("å", "0c0").replace("ø",
"0x0").replace("ä", "0z0").strip()
    #postal_code = re.sub("\D", "", adress_apt)[-4:]
    money = cells[10 * x + 1].text.replace(".", "")
    type_of_sale = cells[10 * x + 2].text[10:]
    year_of_sale = cells[10 * x + 2].text[6:10]
    kr_sqmeter = cells[10 * x + 3].text
    no_of_rooms = cells[10 * x + 4].text
    boligtype = cells[10 * x + 5].text.replace(",", "
").replace("ø", "o").replace("æ", "ae").replace("å",
"a").replace("ä", "a")
    sq_meters = cells[10 * x + 6].text
    build = cells[10 * x + 7].text
    percentage = cells[10 * x + 8].text.strip()

    #print(adress_apt , money, type_of_sale, kr_sqmeter,

```

```
no_of_rooms, boligtype, sq_meters, build, percentage, '\n')
    f.write(address_apt + "," + money + "," + year_of_sale + "," +
type_of_sale + "," + kr_sqmeter + "," + no_of_rooms + "," +
boligtype + "," + sq_meters + "," + build + "," + percentage + "\n")

    x += 1
```

Each cell is scraped to take out the data for: address, price, type of sale, year of sale, square meter price, number of rooms, type of accommodation, size of apartment in square meters, year the building was erected, and percentage of difference in sales price from the original price. Function `f.write()` saves the data into CSV file.

Taking out the address is causing the biggest problems, since Danish language has three special characters: æ, ø, å. In addition to that, it has also been noticed that letters ä, ö and é are present in streets names. The data has to be prepared in a way that allows Power Maps in Excel and Google Maps to identify addresses correctly. If nothing is changed following error occurs:

“UnicodeEncodeError: 'charmap' codec can't encode character '\xf8' in position 32: character maps to <undefined>”

Using the function which replaces errors, gives us following results:

```
>> adress_apt.encode(sys.stdout.encoding, errors='replace')
b'H\xc3\xa4ndelsvej 24| ST. TH2450 K\xc3\xb8benhavn SV'
```

Original address is “Händelsvej 24| ST. TH2450 København SV”. Problem that stems from this solution is that the result of the function is bytes (represented by `b'`), which means that this is not a data of string type. That makes it impossible to pass it into CSV file.

New approach has been decided. Special characters are replaced by string of signs and letters that normally do not exist in words. Afterwards, they are to be replaced manually in excel to the correct letters using find and replace function.

Original letter	Replacement
Æ	0m0
Å	0n0
Ø	0b0
æ	0v0
å	0c0
ø	0x0
ä	0z0

Table 2 - Letter correction table

**Solving problem with
 in text**

The
 break in HTML makes both lines of address combined in output when using .text() function.

Result: Peter Holms Vej 12| 1. TH2450 Kobenhavn SV

The text attribute of a BeautifulSoup tag returns a string composed of all child strings of the tag, concatenated using the default separator (an empty string). To substitute a different separator, one can use the get_text() method.

Therefore break is to be substituted with space using get_text(separator=" ").

Result: Peter Holms Vej 12 1. TH 2450 Kobenhavn SV

Data processed in the following way is prepared to be used by Power Maps and Google Chrome.

5.4. Preparing the dataset for a regression analysis

In this study, the dataset is generated from the real estate database Boliga.dk. The webpage has been scrapped and 54.073 relevant data points has been found. In order to determine the valuation factors that influences the most on the value of a property, 11 different variables have been selected and analysed. The variables are found based on the most used variables in the related literature, and from the theory background.

- **Floor**
Tells on which floor the apartment is located. Ground floor is marked as 0 in the dataset.
- **Price**
Latest sales price.
- **Sales year squared**
Tells the latest sales year of the real estate.
- **Sales meter squared**
Tells the latest sales year of the real estate squared.
- **Sq. Meter price**
Represents the average yearly income, for people living in Copenhagen.
- **Yearly Income**
Represents the average yearly income, for people living in Copenhagen
Tells the sq. meter price based on the latest sales price.
- **No. of Rooms**
Tells the number of rooms in the apartment. Kitchen and bathroom is not considerate as a room according to Danish standards.
- **Interest rate**
Tells the yearly average interest rate level.
- **Sq. Meters**
Amount of sq. meters.
- **Year Built**
Tells the year when the building was build.
- **Population Growth**
Shows the overall population growth in the city of Copenhagen.

These variables are used to conduct the regression analysis. See chapter 6.7. page 59.

Outliers

In a regression analysis, sometimes a few outlying observations can have a deep impact on the estimated coefficients. For this reason, it is important to find and delete these observations from the data set. After running the analysis in SPSS the following output is generated:

Reading the output:

1. The table below shows how the prices are divided into percentage. E.g. The lowest 25% are priced from 1.295.000 DKK and below. 75% of the apartments from our dataset are priced 4.125.440 DKK or below.

		Percentiles						
		5	10	25	50	75	90	95
Weighted Average (Definition 1)	price	740000,00	925000,00	1295000,00	1865000,00	2915000,00	4125440,00	5100000,00
Tukey's Hinges	price			1295000,00	1865000,00	2914680,50		

Table 3 - Output in SPSS

2. Next, we use the 25% upper and 25% lower percentiles to calculate the outliers in the dataset. We find that anything above 6.479.000 DKK could be considered as an outlier. The same goes for all prices below 2.269.000 DKK, but since the dataset does not contain any negative numbers, no outliers will be identified in the lower percentiles. Since the idea of analysis is finding overpriced apartments, it would undermine the purpose of this project to delete the outliers in that matter.

5.5. Data visualisation using Google Maps API

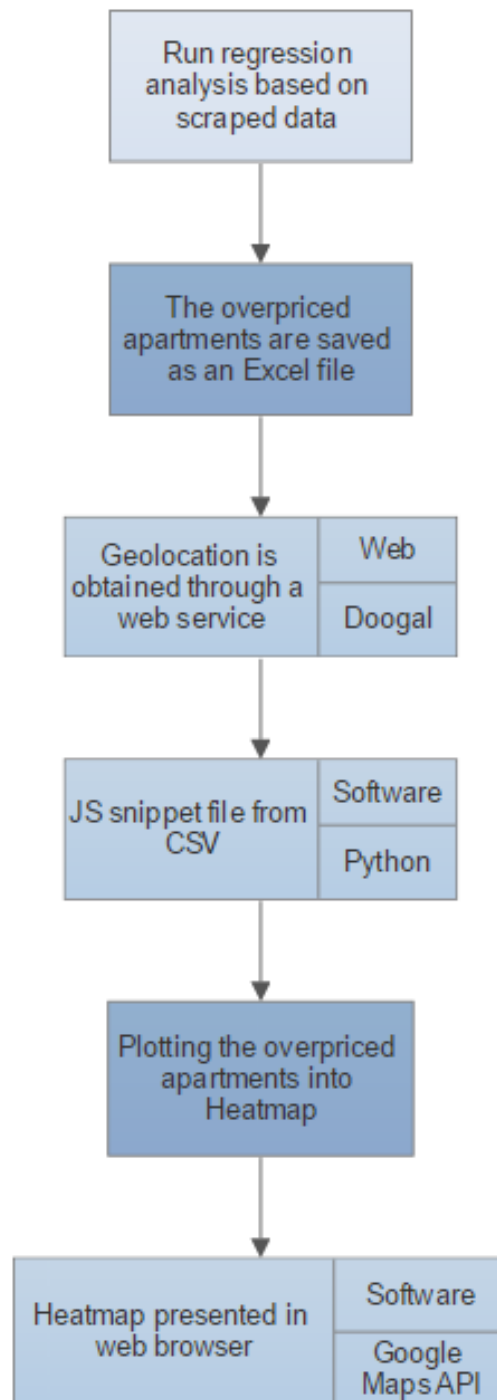


Table 4 - Flowchart of Google Maps API process

Google Maps - Internet service for finding objects, viewing maps, aerial imagery of the Earth, 360 ° panoramic street view (street view), real-time traffic, and route planning by car, public transport, bicycle, foot or ferry. It was created in 2005 by Google. It is free for non-commercial users. Google has created an API that allows you to insert your own map into any web page. Access to the API comes from JavaScript, ActionScript 3 (Google Maps API for Flash) or as a Google Static Maps API. Google Maps API requires a free key that can be obtained by anyone. The key allows access from one domain or domain directory. Google Maps API allows you to integrate a fully functional map with your own data and event handling functions into your website. The service is free until the maps on webpage does not exceed 150,000 requests per 24 hours [24]. User can track the console panel provided by google at <https://console.developers.google.com/>. The console also provides information on libraries available as well API key information.

Obtaining API key

In order to use the Google Maps API, the user is required to obtain a unique key. The key needs to be pasted into the JavaScript code before compiling. The key allows Google to monitor activity of applications developed by user. There is also an extensive tutorial for how to use Google Maps API including examples of importing data into maps, visualizing data, creating legends and drawing on the map.

Heatmap Layer

For the purpose of this thesis a code provided by Google is used. It is freely accessible under URL:

<https://developers.google.com/maps/documentation/javascript/examples/layer-heatmap>

In order to transfer overpriced apartments data into JavaScript the CSV file which contains latitude, longitude and weight (representing how much overpriced an apartment is) needs to be transformed into file that contains correctly encrypted data. In order to pass the locations into heatmap, it needs to be coded in a following manner:

```
{location: new google.maps.LatLng(55.68744, 12.537371), weight:0.04}
```

Function takes latitude and longitude as first two arguments and weight as third.

How to obtain latitude and longitude from full address?

Once the data is obtained and analysis in SPSS is run, we receive data with addresses of apartments and a number that indicates whether they are overpriced. Addresses are transformed into latitude and longitude representation using following geolocation web service: <https://www.doogal.co.uk/BatchGeocoding.php>

That data is pasted into Excel file and subsequently transformed into CSV file. After that process, it is encrypted using “csv2sila.py” (see in appendix).

Once the data is input into the JavaScript and saved it can be presented in the web browser. Results are presented in chapter 6.9.

Explaining csv2sila.py – code that encodes locations for google maps

Code written in python is responsible for encoding the file from CSV to line of code that is read in JavaScript for creating Heatmap. Input of this program is a CSV file that contains values as following: latitude, longitude, weight. Once the program csv2sila.py is executed, a JavaScript snippet file is created. It inputs the data from CSV file into lines that are readable by Google Maps. Following lines of code below define the function createSnippet() which creates a JavaScript snippet:

```
def createSnippet(csvReader, fileName, order):
    # Open file for JavaScript code
    jsSnippet = open(fileName, 'w')

    for row in csvReader:
        jsLine = "{location: new google.maps.LatLng(%s, %s),
weight:%s},\n" % (row['lat'], row['long'], row['weight'])
        jsSnippet.write(jsLine)

    # closing the file
    jsSnippet.close()
```

The main function defines the order in which function arguments are written to the snippet, reads the CSV file and runs the function createSnippet() and saves the snippet in project folder. Full code can be found in Appendix.

Explaining Google Maps Heatmap JavaScript Code

Heatmap is a graphical representation of data where each point is represented by a gradient of colour. It is helpful in representing how dense are the points in the maps, or where the points have the highest value.

The code is a HTML file with JavaScript embedded, that connects to google maps and allows it to be shown in web browser stored on local host.

HTML structure looks as follows:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Heatmaps</title>
    <style>
// Functions that define how does map look in the web browser - size
in //the window, position, height.
    </style>
  </head>
  <body>
    //Function that holds button to toggle Heatmap, change gradient
    etc.
    <script>
    //Map initialization
    //Defining buttons' function, defining gradients
    //Plotting heatmap data using function getPoints()
    </script>
    <script>
    //Line of code that holds the Google Maps API key
    </script>
  </body>
</html>
```

Map initialization:

```
function initMap() {
  map = new google.maps.Map(document.getElementById('map'), {
    zoom: 13,
    center: {lat: 55.683447, lng: 12.591248},
    mapTypeId: 'satellite'
  });
}
```

User defines where is the center of the map on the initial view and how much area does it cover, by toggling the zoom value. Type of map can also be selected. Possible inputs: roadmap, satellite, hybrid and terrain. Satellite seems the most appropriate one, as it allows the viewer to glance upon the building represented by heatmap point.

Buttons definition:

```
function changeOpacity() {  
    heatmap.set('opacity', heatmap.get('opacity') ? null : 0.2);
```

Pressing on the button triggers action which changes the opacity from value 1 to 0.2, which allows user to precisely see building beneath heatmap layer.

Script which includes Google Maps API key:

```
<script async defer  
src="https://maps.googleapis.com/maps/api/js?key=xxxxxxxxxxxx&libraries=visualization&callback=initMap">  
</script>
```

Key number is hidden in order to prevent usage by unwanted persons. Full code can be seen in appendix.

6. Analysis

This chapter provides the reader with a description of how the analysis have been structured and conducted. A brief presentation of Boliga.dk is made before moving on to a flowchart that provides an illustration of the overall process of creating the software from raw data. The regression model in SPSS is explained, and the results are showed in Google Maps API.

6.1. About Boliga.dk

Boliga.dk is a Danish portal for both property seekers and for people generally interested in the housing market in Denmark. The portal is not associated with a particular real estate and allows you to search across the country. It is owned by the Danish online company Euroinvestor. Boliga.dk is visited by approximately 457,000 people a month (pr. November 2015), which makes the 36th most visited Danish web media. Boliga.dk also operates in Sweden and Norway.¹

Boliga today consists of the main page, a related property portal "Selvsalg.dk" (for home sellers who want to sell without the help of a realtor) "Lejnu.dk" (rental) and "itvang.dk" showing foreclosures. Boliga.dk can provide information about the real estate market, such as geographical location, price per square meter, length of stay and previous trade prices. Registered users can put homes they find interesting on their "watch list" and receive alerts when there are changes connected to the specific property. Boliga.dk also associated with a user-controlled housing debate where users discuss topics of interest rate increases to landscapers.

Boliga's is trying hard to promote residential self-sale, by removing the traditional real-estate agencies, and aiming at connecting the seller directly with the buyer. This will save the vendor a lot of expenses. Their goal is to gain a market share of 10%.¹

¹ www.boliga.dk/om

6.2. Flowchart – Overall process

The flowchart communicates the logic of a system in an easy way. It also acts as a blueprint during systems analysis and program development phase.

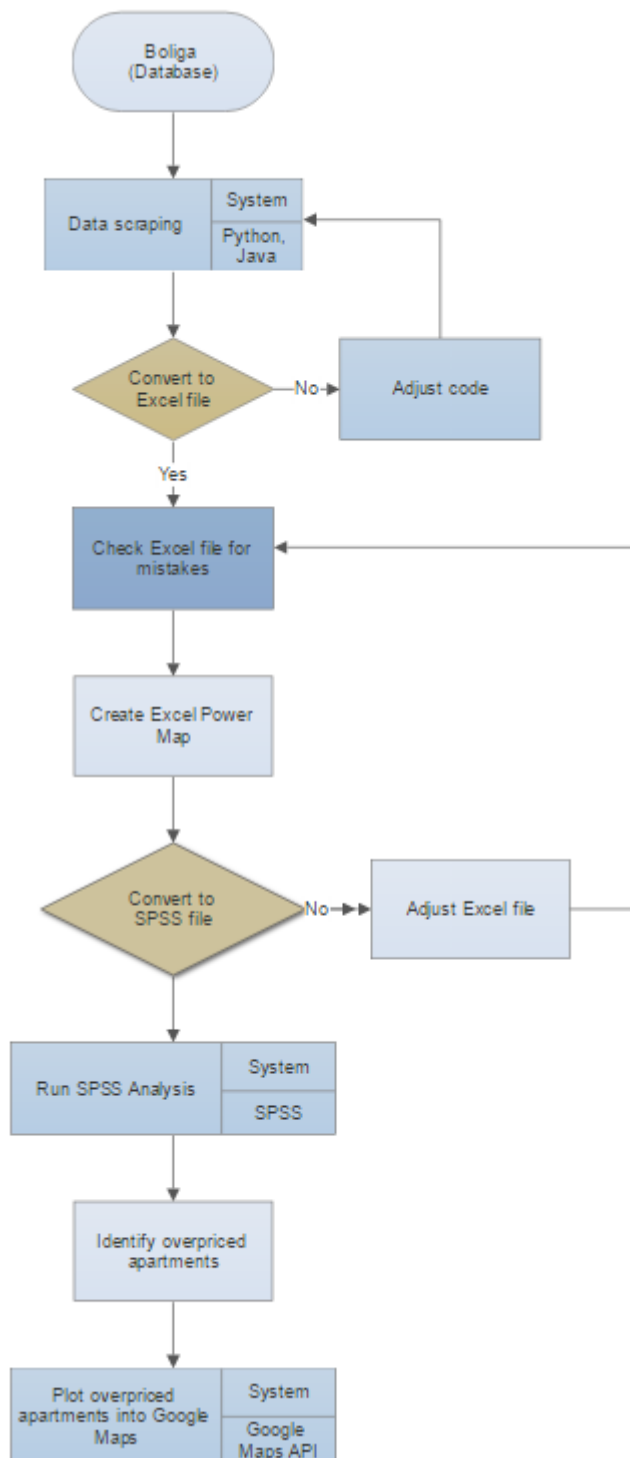


Table 5 - Flowchart of overall process

Maps, to visualize them.

Process:

1. The main data source is Boliga.
2. First, it is essential to convert the raw data into an Excel file. This is done by data scraping Boligas database and converting it into an excel file. This is a very delicate process, in which many adjustments needs to be performed before mastering the code.
3. When the Excel file is finished, the Excel Power map can be created.
4. The Excel file then needs to be converted to an SPSS file in order to run the regression analysis.
5. Next, the overpriced apartments are identified through the regression analysis.
6. Lastly, the overpriced apartments are plotted into Google

6.3. Data scraping results

Data has been scraped with 100% of accuracy. 107.927 were scraped with data of apartments sales in Copenhagen area. After excluding data regarding family sales, house sales, errors etc. 54.073 entries were left for purpose of analysis.

6.4. Visualization - Excel Power Map

Power map is an add-on business intelligence tool for Excel that can be downloaded for free if the user has the MS Office package. Power Map works as a powerful 3D data visualization tool that allows the user to plot geographic and temporal data onto custom maps. Based on geocoding it allows the user to pinpoint a geographic location based on a description E.g. Address, city, postal code, country etc. If installed successfully the plugin will appear in Excel under the “INSERT” tab.

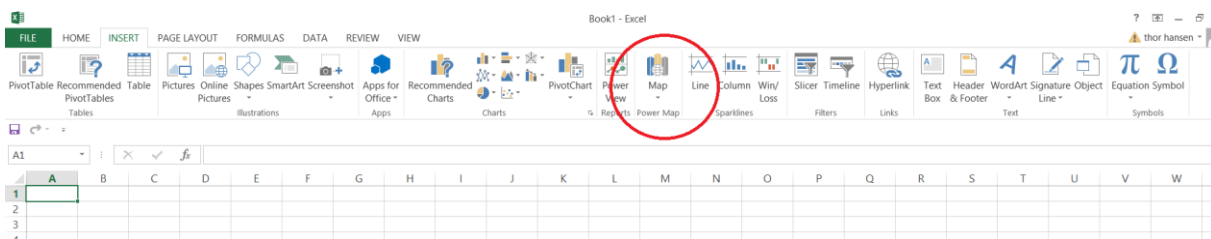


Figure 11 - Excel Power Maps

6.4.1. Preparing the data

The data set should be structured into an Excel table format, where each row represents a unique record. This is important to allow Power Map to interpret it correctly when it plots the geographic coordinates.

	A	B	C	D	E	F	G	H	I
1	adress	price	year_of_sale	type_of_sale	sq_m_price	rooms	Type_of_aprt	size[sq_m]	year_of_built
2	Margretheholmsvej 68 1. TV 1432 København	4025000	2017	Alm. Salg	43.279	4	Lejlighed	93	2014
3	Nansensgade 96 4. TH 1366 København	2663000	2017	Alm. Salg	35.039	3	Lejlighed	76	1878
4	Fredericiagade 21D 1310 København	2430000	2017	Alm. Salg	34.714	2	Lejlighed	70	1968
5	Torvegade 25 2. TV 1400 København	2925000	2017	Alm. Salg	26.590	4	Lejlighed	110	1933
6	Sankt Pauls Gade 3 3. TV 1313 København	2895000	2017	Alm. Salg	46.693	2	Lejlighed	62	1875
7	Frederiksberggade 25C 2. TH 1459 København	6700000	2017	Alm. Salg	26.800	4	Lejlighed	250	1900
8	Klerkegade 17B ST. TV 1308 København	3750000	2017	Alm. Salg	52.083	3	Lejlighed	72	1810

Figure 12 - Data in Excel Power Maps

Power Map requires one or more geographic values per row of data. This value could be a City, Country, State or Address. The more specific the geographic values is described the more accurate Power Map will plot the data.

The data is structured into 10 different rows. First row represents the specific address including the postal code and city. Next row represents the listed price upon the time where the condominium was set for sale. The following labels needs no further explanation.

Select data

Next step is to select the data, which in this case is the whole table, then we click on “INSERT”, “Maps” and “Launch Power map”. The Power Map window will appear, and the actual visualisation work can begin.

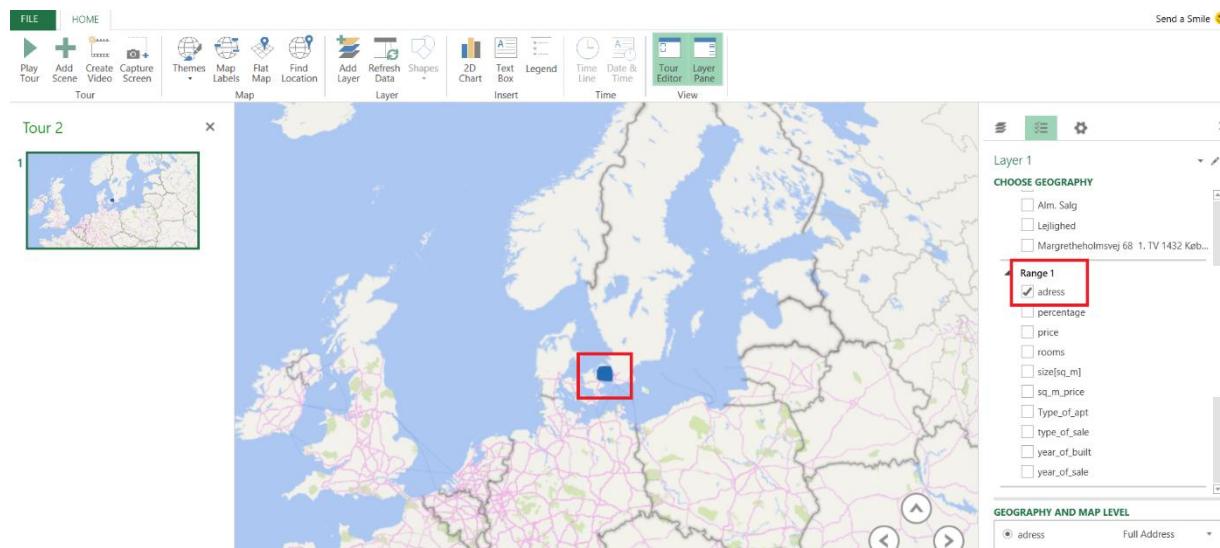


Figure 13 - Plotting data into Excel Power Maps

The plugin will automatically detect one of the parameters “Address” and show the exact location on the map.

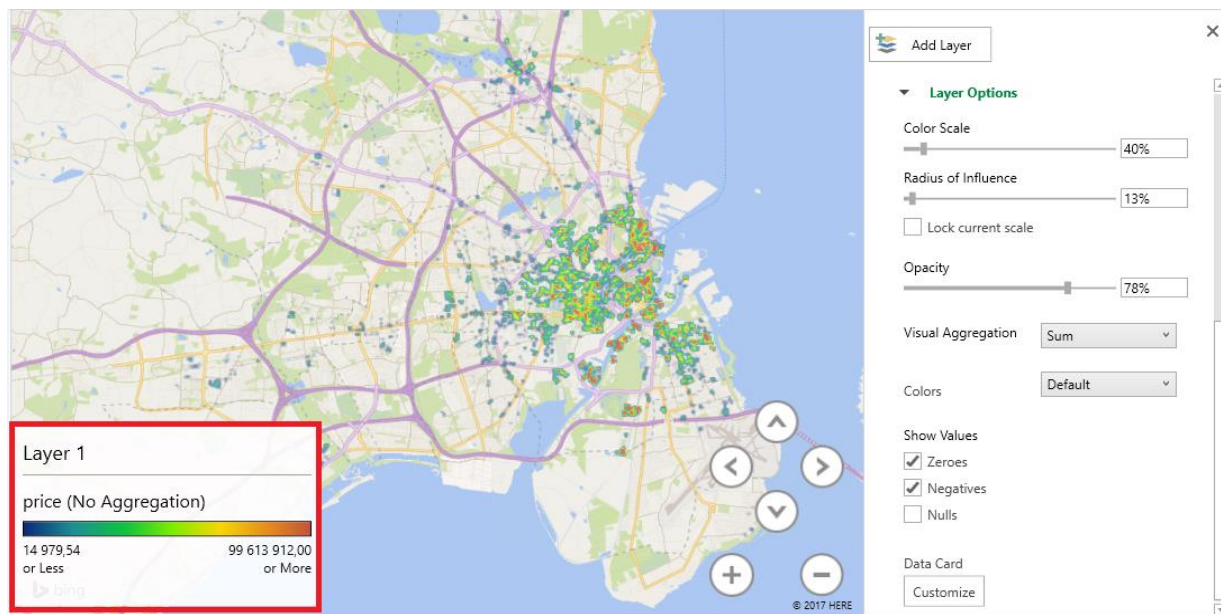


Figure 14 - Output

Visualization

Power Map supports five types of visualizations:

- Stacked Columns
- Clustered Columns
- Bubble
- Heat Map
- Region

These options can be selected in the Layer Pane. To visualize the price range throughout Copenhagen the Heat Map option was used. Heat map gives an easy and quick overview of how the price changes. The scale goes from blue to red. It is possible to adjust the colour scale, the radius of influence and the opacity.

Errors with Power Map

The data set used consist of 107.927 unique entries scraped from Boliga.dk from selected postal codes for Greater Copenhagen area. The data set represents every sold condominium and house from 2006-2017. From that whole of data, only condominium sales will be represented in this paper. Also, family sales will be excluded as they are sold on a price 15% lower than the market price. After excluding aforementioned data, 58.707 unique entries have been noted.

From the 54.073 entries Power Map localizes 74% and marks 26% as errors. These errors consist of addresses that Power Map is not able to localize or addresses that Power Map localizes but marks as a “guess” meaning the addresses is not 100% correct according to the database. This means that 18,7% is actually plotted on the map but Power Map just is not positive whether the marked area is 100% correct. The last 7,3% is completely missing from the map. A wide range of issues affects the end result:

- Power map does not recognize new postal codes, such as 2150 Nordhavn. Other postal codes such as 2500 Valby are not recognized by Power Map. Changing the name of the city, in this case Valby, to Copenhagen will make Power Map localize the address.
- Power Map can't read the specific details of an address, such as “TV or ST” that indicates whether is the right or left condominium
- Power map is unable to localize certain addresses without any explanation. This effects 7.3% of the entries

6.5. Data cleaning

When working with data it is always essential to perform error detection. No matter how efficient the process of data entry is, errors will still arise. For this reason data validation and correction is highly important. One important aspect of data cleaning is the identification of the root cause of the errors detected and using that information to improve and prevent the errors from reoccurring [25].

Errors found

Data cleaning is basically centre around improving the quality of data. Errors in data are very common and to be expected. The errors in the dataset obtained from Boliga.dk are basically identified as being priced incorrectly. For instance, the picture below represents some of the apartments that are clearly priced incorrectly in-correctly.

	A	B	C	D	E	F	G	H	I	J	K
1	H.C. ørstedes Vej 63 1. 03 1879 Frederiksberg C	33371000	2016	Alm. Salg	953457	1	Lejlighed	35	2016		
2	Vodroffsvej 16 ST. MF 1900 Frederiksberg C	26200000	2010	Alm. Salg	270103	2	Lejlighed	97	1995		
3	Prins Constantins Vej 8 ST. TV 2000 Frederiksberg	19200000	2014	Alm. Salg	249350	3	Lejlighed	77	1884		
4	Smallegade 20B 4. TH 2000 Frederiksberg	24200000	2012	Alm. Salg	232692	5	Lejlighed	104	1900		
5	Åboulevard 53 4. TH 1960 Frederiksberg C	8300000	2014	Alm. Salg	207500	1	Lejlighed	40	1986		
6	Pile Alle 14A 2000 Frederiksberg	8500000	2006	Alm. Salg	188888	2	Lejlighed	45	1802		
7	Jakob Dannefærds Vej 6B ST. TV 1973 Frederiksberg C	15251000	2008	Alm. Salg	152510	4	Lejlighed	100	1885		
8	Thorvaldsensvej 3 2 1871 Frederiksberg C	23500000	2015	Alm. Salg	127027	3	Lejlighed	185	1870		
9	Niels Ebbesens Vej 19 1 1911 Frederiksberg C	9500000	2014	Alm. Salg	114457	3	Lejlighed	83	1871		
10	H.C. ørstedes Vej 30 1 1879 Frederiksberg C	10000000	2008	Alm. Salg	88495	5	Lejlighed	113	1850		
11	Alhambravej 13 1 1826 Frederiksberg C	7700000	2006	Alm. Salg	86516	2	Lejlighed	89	1873		
12	Alhambravej 18 1. TH 1826 Frederiksberg C	5200000	2014	Alm. Salg	71232	3	Lejlighed	73	1874		

Table 6 - Data cleaning output

Column B indicates the sales price. The first apartment is priced at 33.337.100. DKK for a 35 m² apartment in Frederiksberg. This is obviously a mistake that has occurred from Boliga.dk. 102 apartments was deleted due to troubleshooting. This was done manually by looking up suspiciously cheap/expensive apartments. After removing the errors from the dataset 54.073 datapoints were left.

6.6. Variables in SPSS

This section presents the different variables used in the regression analysis.

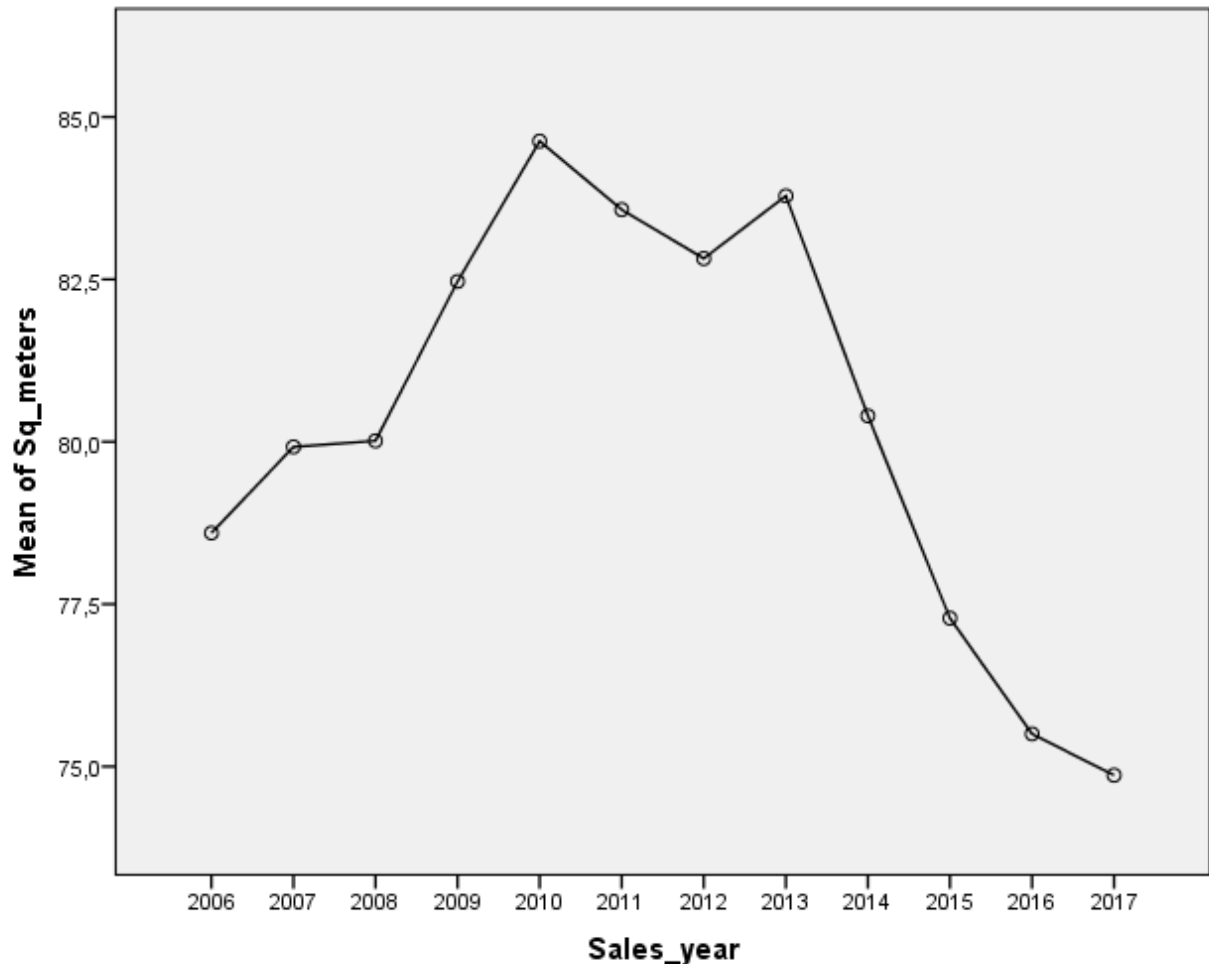


Figure 15: Means plot from SPSS analysis

Mean plots are used to see if the mean varies between different groups of data. Mean plot can also be used with ungrouped data to determine if the mean is changing over time. In this case two parameters have been selected: Mean of Square-meters and Sales-Year. What we can conclude from the plot is following:

- From the years 2006-2009/2010 the tendency clearly indicates that the sale of larger apartments was increasing.
- From 2010 until today the trend line turned the other way and began decreasing. Today it is smaller apartments that are being sold.

Scatter plot

The scatter plot allows visualization of how the apartments in our dataset are distributed. The price of the apartment is represented as Y and size as X. It is clearly that most apartments lay in the price field between 0-10.000.000 DKK. The size of the apartments are concentrated between 0-200 m². The scatter plot gives a quick overview of the two variables. It is important to stretch that the scatter plot can be a bit misleading, for example it might appear as there is a considerable amount of apartments above 100 m², even though this actually only accounts for 20% (11.006 apt) of the apartments from the dataset. In other words 80% (43067 apt) of the apartments from the dataset are below 100

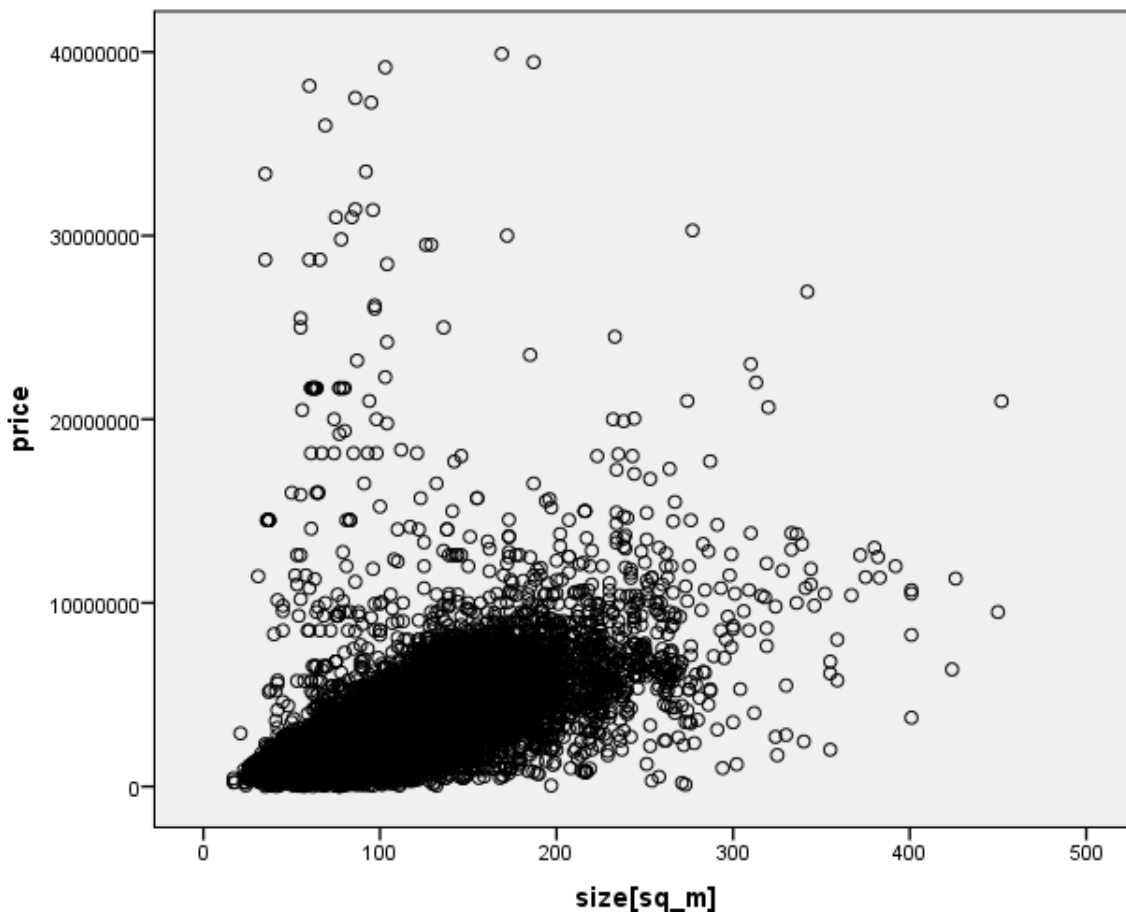


Figure 16: Scatter plot from SPSS analysis

6.7. Predicting Values of Dependent Variables

To run a multiple regression analysis the first thing to do, is select the variable that is to be predicted (the dependent variable) from more than one independent variable, by using multiple regression analysis.

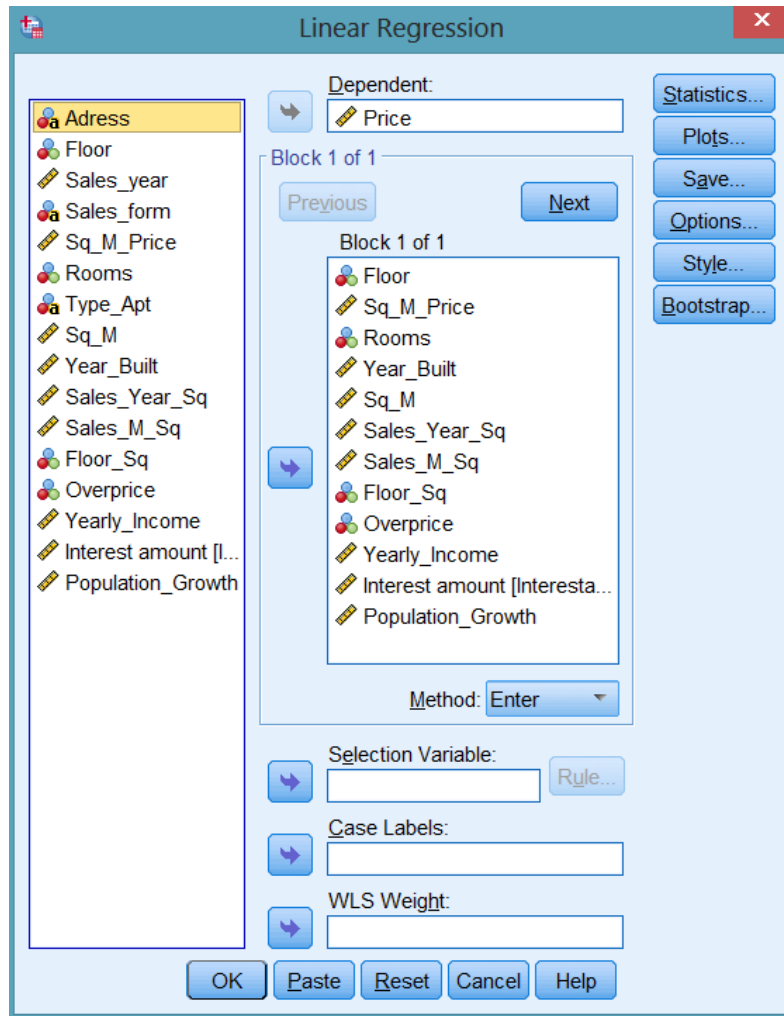


Figure 17: Variables in SPSS

From the regression analysis in SPSS we get the following results:

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-11202372,6	160981,078		-69,588	,000
	Floor	-9746,607	1405,456	-,015	-6,935	,000
	Sq_M_Price	67,745	,290	,467	233,301	,000
	Rooms	-22328,546	1672,537	-,020	-13,350	,000
	Year_Built	240,084	24,354	,009	9,858	,000
	Sq_M	18749,260	153,063	,500	122,494	,000
	Sales_Year_Sq	-166,342	17,077	-,011	-9,740	,000
	Sales_M_Sq	16,207	,557	,094	29,077	,000
	Floor_Sq	1419,952	217,171	,013	6,538	,000
	Overprice	-157418,617	3194,803	-,066	-49,273	,000
	Yearly_Income	28,896	,542	,121	53,314	,000
	Interest amount	9,465	,067	,273	141,011	,000
	Population_Growth	98,202	2,712	,044	36,215	,000

a. Dependent Variable: Price

Figure 18: SPSS output

R-Square provides an indication of the explanatory power of the regression model. What would be considered a good R-Square result depends on the setting and type of data used. R-Square simply tells the percentage of variance in the depend variable explained by the collection of independent variables. The R-Square value range from 0 to 1, where 1 represents a perfect correlation between the independent and depend variable(s).

If a significant level can be accepted as -0,10-0.050 most of the variables in the regression model have a significant impact on the dependent variable, except 'Floor', 'Floor_Sq', 'Overprice', 'Sales_Year_Sq', 'Rooms' and 'Population_Growth', for this reason the variables are removed from the analysis and the regression is conducted again. This time the model only has significant coefficients.

Coefficients ^a						
		Unstandardized Coefficients		Standardized Coefficients		
Model		B	Std. Error	Beta	t	Sig.
1	(Constant)	-8247780,451	129865,615		-63,510	,000
	Sq_M_Price	70,467	,218	,486	322,871	,000
	Sq_M	18473,113	127,124	,492	145,316	,000
	Year_Built	335,044	24,331	,012	13,770	,000
	Sales_M_Sq	17,452	,548	,102	31,840	,000
	Yearly_Income	20,985	,416	,088	50,421	,000
	Interest amount	7,992	,064	,231	124,557	,000

a. Dependent Variable: Price

Figure 19 - Most significant variables

The regression analysis is now conducted with the most significant variables. The following variables have the highest positive correlation:

- Square meter (.492)
- Sq_M_Price (.486)
- Interest rate (.231)
- Yearly Income (.088)

There is no surprise in these results. They align well with our expectations. The square meter price is the most significant variable. The higher amount of square meters equals a higher price. The second most significant variable is the square meter price, which is quite logical, since the higher the square meter price is, the higher the total price will be. The national interest rate is also a key driver in relation to the real estate prices. The regression analysis tells, that when the interest rate goes up, the general real estate prices goes down. This aligns well with the statistical material that exists in this field. See appendix for further explanations. Lastly, the 'Yearly Income' has a positive correlation as well. Basically, when the general incomes raises so does the general real estate market. ²

The only real surprise, is that the coefficient 'Population_Growth' has a very low significance. One way of explaining why this coefficient is insignificant, is because of the relatively small changes in the population growth. From 2006-2017 the difference

is 16% in the population growth, and compared to the interest rate and changes in the income level, the change is low.²

6.7.1. Dummy variables

Dummy variables were used in the regression analysis but did not change the overall output. The dummy variable was set for identifying the overpriced apartments from the regular ones. The results were insignificant and for that reason we chose to leave them out of the analysis. See appendix for results

6.7.2. Model summary

The model summary tells how accurate our model, consisting of the variables mentioned before, correlates with the price. The R-Square value is 0,799 which equals to 79% accuracy. This result is highly satisfying and proves that the model in this paper has a good fit.

Model Summary				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	,894 ^a	,799	,799	508761,195

a. Predictors: (Constant), Sales_year, Sales_M_Sq, Yearly_Income

Figure 20 - Significance of variables

² <https://www.kk.dk/sites/default/files/Status%20p%C3%A5%20K%C3%B8benhavn%202016.pdf>

6.8. Conclusion of SPSS analysis results

When constructing a regression model it is important to have the right balance, between which variables should be used in the model and what data is available. It was decided to use 11 variables from the beginning, based on relevant theory and related work. Of those variables, only six showed to be significant.

Based on the results from this chapter the following can be stated:

- 75% of the apartments in the dataset are priced at 4.100.000 DKK or below.
- All apartments above roughly 6.400.000 DKK could be considered as outliers, we still chose to use them in the analysis in order to identify the extremely overpriced apartments.
- Six significant variables are found to have an impact on the general price level of the apartments in Copenhagen.
- The size of an average apartment being sold, is considerably lower today than back in 2010, 80% of all the sold apartments are 100 m² or below.

6.9. Identifying overpriced apartments

To find the overpriced apartments it is necessary to find the predicted prices first. This is done through a prediction analysis in SPSS:

1. First calculate the difference between the actual prices and the predicted prices.

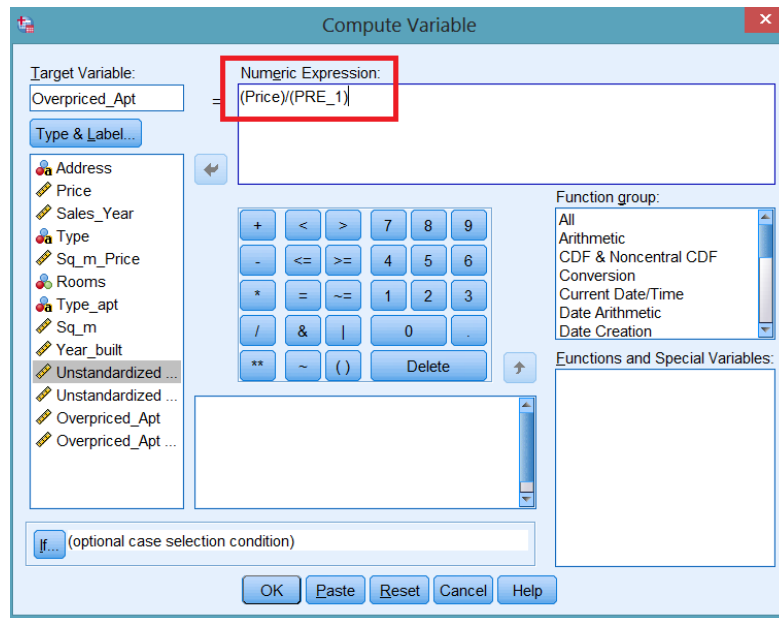


Figure 21 – Computing variable for finding overpriced apartments

2. The result obtained from this is a new variable that shows the disparity between actual price and predicted price. The difference is found simply by dividing the actual price with the predicted price, as shown above.
3. The results obtained from the previous steps are presented in the highlighted column on the right, in the table below. If the number is above 1.00, the apartment is over-priced, below 1.00 the price is under-priced. For instance, if the price is set at 1.45, this equals to the apartment being overpriced by 45% according to the predicted price.




 PRE_1	 RES_1	 Overpriced_Apt
8210278,78852	-510278,78852	,94
6151097,45243	-951097,45243	,85
6378048,51495	-583048,51495	,91
7881418,28349	1393581,71651	1,18
7573927,31591	1176072,68409	1,16
5982903,20891	-682904,20891	,89
7624084,44351	1360915,55649	1,18
8043054,21072	1956945,78928	1,24
6332301,41510	198698,58490	1,03
6427475,78951	472524,21049	1,07

Table 7 - SPSS data

4. Next, we categorize the overpriced apartments into city districts. Following results are obtained:

Overall overpriced Apartments – Central Copenhagen		
Inner city + Vesterbro	Østerbro	Frederiksberg
5.6%	5.1%	4.8%

Table 8 - Overpriced apartments CPH

5. The table above reflects the distribution of overpriced apartments in each district. Inner city + Vesterbro are affected by a 5.6% overprice in general.

Average of overpriced Apartments only – Central Copenhagen		
Inner city + Vesterbro	Østerbro	Frederiksberg
32.9%	19,8%	17,8%

Table 9 - Overpriced apartments CPH

6. The table above accounts for the average price of the isolated overpriced apartments. For example, of all the overpriced apartments in 'Østerbro' the average price is 19.8% above predicted price.

6.10. Plotting the apartments in Google Maps

The results obtained from the previous sections are then plotted into Google Maps. The purpose of this is to research whether there is any patterns or somehow a correlation for the overpriced apartments.

Østerbro

In Østerbro 428 are found to be overpriced by 20% and above. Out of the 428 overpriced apartments 97 of them was found to be overpriced by 50% and above.



Figure 22 - Google Maps API output

Inner city + Vesterbro

Inner city and Vesterbro accounts for 619 apartments being 20%+ overpriced. 140 of the overpriced apartments accounts for being overpriced by 50% and above. Especially the areas in inner Vesterbro and parts of Christianshavn are effected by being overpriced. Surprisingly enough the most overpriced apartments are not located in the absolute city centre, but in parts of Christianshavn, one explanation for this could be that the prices by default are priced extraordinary high in the city centre.

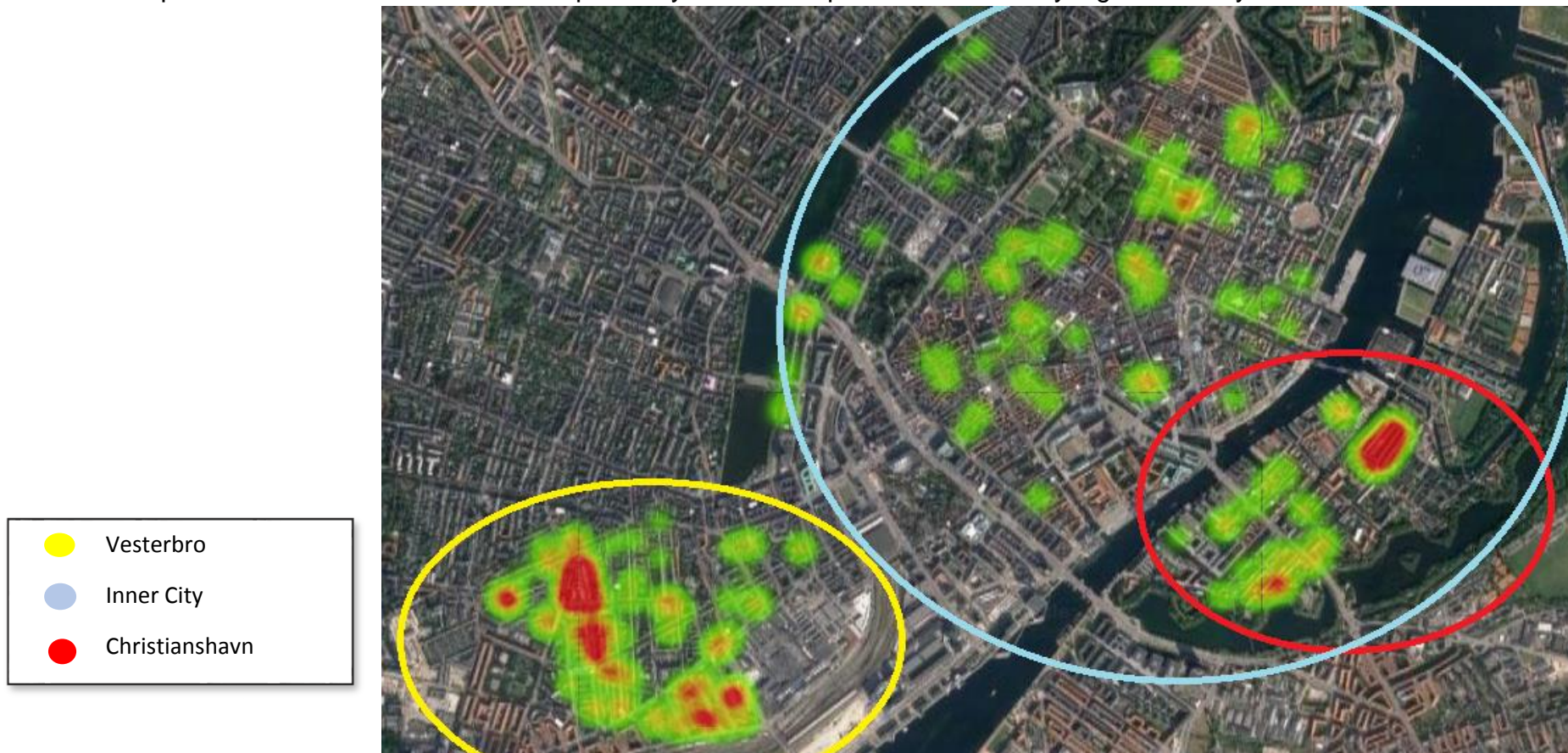


Figure 23 - Google Maps API Output

Frederiksberg

In Frederiksberg 343 apartments are overpriced by 20% and above. 77. Apartments are overpriced by 50% and above.



Figure 24 - Google Maps API Output

6.11. Section summary

Many factors influences the areas with a high concentration of overpriced apartments however, three main factors appears to play a key role in regards of overpriced apartments:

- Newly build buildings
- Buildings with a sea view
- Buildings placed in urban renewal zones

One of the clearly overpriced areas in Østerbro, is concentrated around the waterfront, this could easily explain why the prices are so high. Another major factor that appears to influence the prices are newly build apartments. Newly build condominiums makes up for a big part of the overpriced areas. This pattern shows in all the city districts. Furthermore, some of the overpriced areas are located in urban renewal areas. For example, the area “Klimakvarteret” in Østerbro is a project founded by the City of Copenhagen, with the purpose of creating a neighborhood geared to deal with the climate changes that the future brings, such as the large rainfalls that Denmark has suffered from in the event of a cloudburst [26]. Also the area in Vesterbro named ‘Central Vesterbro’ has experienced a massive urban renewal in the period 2011-2017, which could account for the extraordinary high prices in this exact area.

7. Conclusion

The last chapter of this project, ties up the whole thesis with a conclusion that leads to future work. Based on the selected method, theory and analysis we try to address the issue:

Is it possible to identify overpriced apartments through data analysis in the greater Copenhagen area?

The theoretical background assures the dissertation a high validity and reliability. It gives a profound insight in the complex mechanisms that drives the real estate market. Based on the findings from the theory, different conclusions are made and put to a test in the analysis section.

From the analysis part it is concluded that we have managed to exploit all feasible information from boliga.dk database. A linear regression model has been developed and executed. From the results of the regression analysis, we can conclude that the model is valid up to an accuracy of 79%. Regarding the signification of our model we can conclude that six of the original eleven variables are indeed significant. Our results were aligned with the general belief, that there is a close relationship between socio-economic variables such as the interest rate and income level, and the property prices.

Our results and approach, from Google Maps API, provide real-estate agencies with a novel way to find over-priced apartments. There appear to be a close relationship between newly constructed areas and urban renewal projects, and overpriced apartments.

We developed a working prototype of an application, and the results are highly satisfying. The results are encouraging enough to warrant further work on the matter.

8. Criticism & Future work

The criticism mainly focuses on the overall perspective of the issue in terms of differences between a theoretical and practical approach to create new software, hereafter the focus is on the scientific sources and the related work.

The analysis is characterized by having a practical approach that not necessarily match the theory in the field. By only having, a limited access to the required software, in the sense that much software such as Google Maps API requires a payment subscription to be able to work with it profoundly, sets a natural limit in regards of the results. The conclusions that are obtained from the related work are based on scientific papers from different parts of the world. This could be contradictory to the Danish conditions in the real estate market, since major economic, demographic and political factors might make the variables hard to transfer directly to the Danish real estate market. Other important parameters such as, if the building has been renovated or the distance to public transportation could affect the price. However, we feel confident about the results. The variables are objective and transparent, and reflects the conditions in the real estate market in Copenhagen in the period 2006-2017, based on the free accessible data that exist in this field.

8.1. Future work

The future goal is to create a working application for phones with the data storage and all of the calculations executed in cloud technology. It would allow user to zoom in on a certain area of Copenhagen and see what are the prices of apartments sold, the development of the prices for a given period of time, as well as identifying an exact building and showing information such as sales records within the recent years. Additional information such as how much does the floor level affect the square meter price in that building would be presented.

It requires a tighter integration of systems working together, as well as periodical scraping that allows the data to update automatically. Furthermore, we advise to create an interface where the user can exclude uninteresting information (such as family sales, different types of accommodation etc.).

9. Bibliography

- [1] Saunders, Mark, Philip Lewis, and Adrian Thornhill. *Research Methods For Business Students*. 1st ed. Harlow [etc.]: Pearson Education, 2007. Print.
- [2] Sadolin, Albæk. *Copenhagen Property market report*, 2016.
- [3] Befolkning og Befolkningsfremskrivning. N.p., 2017. Web. 21 Mar. 2017. Available at: <http://www.dst.dk/da/Statistik/emner/befolkning-og-valg/befolkning-og-befolkningsfremskrivning>
- [4] Eurostat - Tables, Graphs and Maps Interface (TGM) Table. *Ec.europa.eu*. N.p., 2017. Web. 15 Mar. 2017. Available at: <http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec00114&plugin=1>
- [5] Danmarks Nationalbanks Rentesatser, Pengemarkedsrentesatser Samt Obligationsrentegennemsnit Ultimo (Pct Pa) Efter Type - Statistikbanken - Data Og Tal". *Statistikbanken.dk*. N.p., 2017. Web. 10 May 2017. Available at: <https://www.statistikbanken.dk/statbank5a/SelectVarVal/Define.asp?Maintable=MPK3&PLanguage=0>
- [6] Miles, Matthew B, A. Michael Huberman, and Johnny Saldaña. *Qualitative Data Analysis*. 1st ed. Thousand Oaks: Sage Publications, 2014. Print.
- [7] Saunders, Mark, Philip Lewis, and Adrian Thornhill. *Research Methods For Business Students*. 1st ed. Harlow [etc.]: Pearson Education, 2007. Print.
- [8] Boelhouwer, Peter. *Home Ownership*. 1st ed. Delft: DUP Science, 2005. Print.
- [9] Blixen, Finecke. Frederik. *House-price effects on the economy and households*. CBS, 2010
- [10] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. John Wiley and Sons, 2012

- [11] "Unusual Traffic From Your Computer Network" - Google Search Help". *Support.google.com*. N.p., 2017. Web. 30 May 2017.
<https://support.google.com/websearch/answer/86640?hl=en>
- [12] A. Arasu and H. Garcia-Molina, *Extracting Structured Data from Web Pages*, Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 337-348, 2003.
- [13] Hirschey, Jeffrey. *Symbiotic Relationships: Pragmatic Acceptance of Data Scraping*. SSRN Electronic Journal
- [14] „Googlebot – Search Console – Help”. N.P. 2017. Available at:
<https://support.google.com/webmasters/answer/182072?hl=pl> [6 May 2017]
- [15] “Google Webmasters – Support, Learn, Connect & Search Console” N.P. 2017. Available at:
https://www.google.com/webmasters/#?modal_active=none [21 May 2017]
- [16] “Orbitz sued by Southwest Airlines”. N.P., 2017 Available at:
<http://www.ecommercetimes.com/story/9518.html> [09 April 2017]
- [17] See: Facebook, Inc v. Power Ventures. 844 F.Supp.2d 1025 (E.D. Cal. 2012)
- [18] See: Daniel J. Gervais, *The Protection of Databases*, 92 CHI.-Kent L. Rev. 1109 (2007)
- [19] See: Intel Corp. v. Hamidi, 71 P.3d 296 (Cal. 2003)
- [20] Chen, H., Chiang, Roger H. L., Storey, Veda C. *Business Intelligence and Analytics: From Big Data to Big Impact*. MIS Quarterly Vol. 36 No. 4/December 2012
- [21] Glez-Pena, Daniel. *Web Scraping Technologies in an API World*. Briefings in Bioinformatics 15.5 (2013): 788-797.
- [22] Hipp, J., Wirth, R. (2008) DamilerChrysler Reasearch & Technology FT3/KL, Wilhlem-Shickard-Institute, University of Tuingen – *CRISP-DM: Towards a Standrad Process Model for Data Mining*

- [23] *15 Python Libraries for Data Science*. Hiring | Upwork. N.P., 2017. Available at: <https://www.upwork.com/hiring/data/15-python-libraries-data-science/> [12 April 2017]
- [24] "Usage Limits | Google Places API for Android | Google Developers". Google Developers. N.P., 2017. Available at: <https://developers.google.com/places/android-api/usage#enable-billing> [12 April 2017]
- [25] Hellerstein, Jospeh, *Quantitative Data Cleaning for Large Databases*, 2008
- [26] KLIMAKVARTER ØSTERBRO: ("Klimakvarter Østerbro") *Klimakvarter.dk*. N.p., 2017. Web. 8 Mar. 2017
- [27] "Optimizing Quota Usage When Geocoding | Google Maps Geocoding API | Google Developers". *Google Developers*. N.p., 2017. Available at: <https://developers.google.com/maps/documentation/geocoding/geocoding-strategies> [11 May 2017]
- [28] Hu, Robert, Sjogren, Emil. *Analysis and prediction of apartment prices in inner city Stockholm*. Department of Mathematical Statistics, KTH Engineering Sciences, 2014.
- [29] Lowrance, Roy E. *Predicting the market value of single-family residential real estate*. Department of Computer Science, New York University, 2015.
- [30] Ng, Aaron. *Machine learning for a London housing price prediction mobile application*. Department of Computing, Imperial College London. 2014
- [31] Candas, Ezgi, Bagdatli, Seda, Yomralioglu, Tahsin. *Determining the factors affecting housing prices*. Department of Geomatics Engineering, Istanbul Technical University, 2015.