Coding Project - 2

**Introduction to Programming**

PROG1000-1 19V

**Spring 2019**

**Handelshøgskolen v/ OsloMet**

**Oslo Metropolitan University**

Python Friends:

David An Tran Pham

Thomas Nielsen

Peter Marius Stangeland

Tormod Hauane

Content:

[1 Introduction 3](#_Toc5213179)

[1.1 Jupyter Notebook 3](#_Toc5213180)

[2 Python to analyze 4](#_Toc5213181)

[2.1 Importing packages 4](#_Toc5213182)

[2.2 Pandas as pd. 4](#_Toc5213183)

[2.3 Matplotlib.plyplot as plt. 4](#_Toc5213184)

[2.4 Numpy as np. 5](#_Toc5213185)

[2.5 From Sklearn import linear\_model as lm 5](#_Toc5213186)

[2.5.1 from Sklearn.model\_selection import train\_test\_split as tts 5](#_Toc5213187)

[2.6 Statsmodel as sm 5](#_Toc5213188)

[3 Reading files and analyzing data 6](#_Toc5213189)

[3.1 Distribution of number of rooms 6](#_Toc5213190)

[3.2 Analytical observations 6](#_Toc5213191)

[3.3 House price analysis 7](#_Toc5213192)

[3.3.1 Coefficient 8](#_Toc5213193)

[3.3.2 Estimation 8](#_Toc5213194)

[3.3.3 R-square 9](#_Toc5213195)

[3.3.4 Prediction vs. reality 9](#_Toc5213196)

[3.3.5 Trend line 10](#_Toc5213197)

[3.3.6 Correlation 11](#_Toc5213198)

[3.4 Multiple analysis 11](#_Toc5213199)

[4 Simple linear regression coding 14](#_Toc5213200)

[4.1 Using matrix 14](#_Toc5213201)

[4.2 Using Sklearn 15](#_Toc5213202)

[4.3 Using Statsmodel 16](#_Toc5213203)

[5 Multiple Regression coding 17](#_Toc5213204)

[5.1 Using matrix 17](#_Toc5213205)

[5.2 Using Sklearn 18](#_Toc5213206)

[5.3 Using Statsmodel 18](#_Toc5213207)

[6 Comparison 19](#_Toc5213208)

[6.1 Matrix vs. Sklearn vs. Statsmodel 19](#_Toc5213209)

[6.1.1 Matrix 19](#_Toc5213210)

[6.1.2 Sklearn 19](#_Toc5213211)

[6.1.3 Statsmodel 19](#_Toc5213212)

[6.2 Project-1 and Project-2 19](#_Toc5213213)

[7 Machine learning 21](#_Toc5213214)

[7.1 Testing and Training 30% 21](#_Toc5213215)

[7.2 Testing and Training 50% 22](#_Toc5213216)

# 1 Introduction

In PROG1000 Coding Project-2 we were given an excel (.xlsx) file containing the housing prices for 100 apartments in Oslo, a given day in December 2018. The file contained five independent variables and one dependent variable. The project was about making a python program by using Jupyter Notebook, and to analyze the housing market in Oslo. Secondly, we will use Machine Learning for prediction.

In Coding Project- 2 we will provide you both simple and multiple linear regression. In addition, we will compare the results by using matrix, Statsmodel and Sklearn. This in order to find weaknesses and strengths within the different methods.

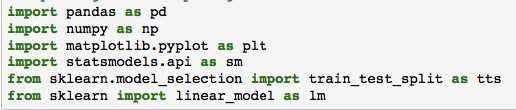
## Jupyter Notebook

Jupyter Notebook is an interface that allows us to program in a web browser. Jupyter is a digital notebook that gives us the ability to execute commands and draw charts. We used Jupyter to organize data and make statistical models.

# 2 Python to analyze

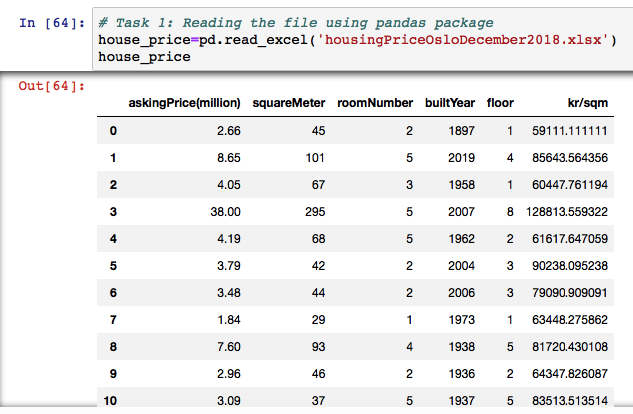
## 2.1 Importing packages

To perform house price analysis, we imported these relevant packages:



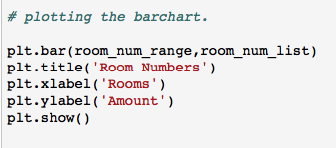
## 2.2 Pandas as pd.

Necessary to read external data files. At the same time, the program allows for simple analysis, data structuring and other tools.



## 2.3 Matplotlib.plyplot as plt.

To visualize the data, we imported a package in python called “matplotlib”. This package makes it possible to generate different types of diagrams; like histograms, bar charts and scatter plots.



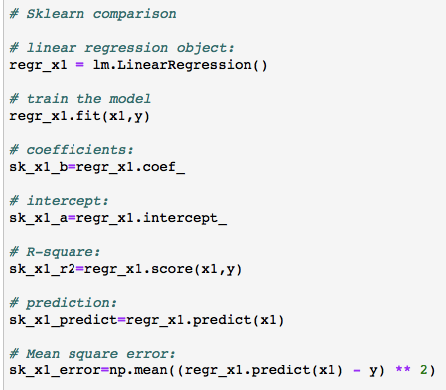
## 2.4 Numpy as np.

Numpy is a fundamental package containing many necessary functions such as array, matrix, mean and so on.



## 2.5 From Sklearn import linear\_model as lm

Sklearn is a package that gives you simple and efficient tools to perform data analysis.



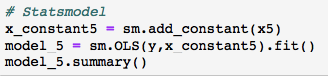
### 2.5.1 from Sklearn.model\_selection import train\_test\_split as tts

To perform machine learning in Python, the Sklearn package is necessary.



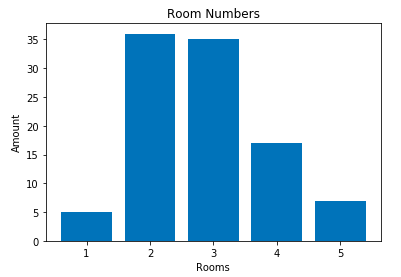
## 2.6 Statsmodel as sm

Statsmodel provides us a statistical model that we can use to analyze data.



# 3 Reading files and analyzing data

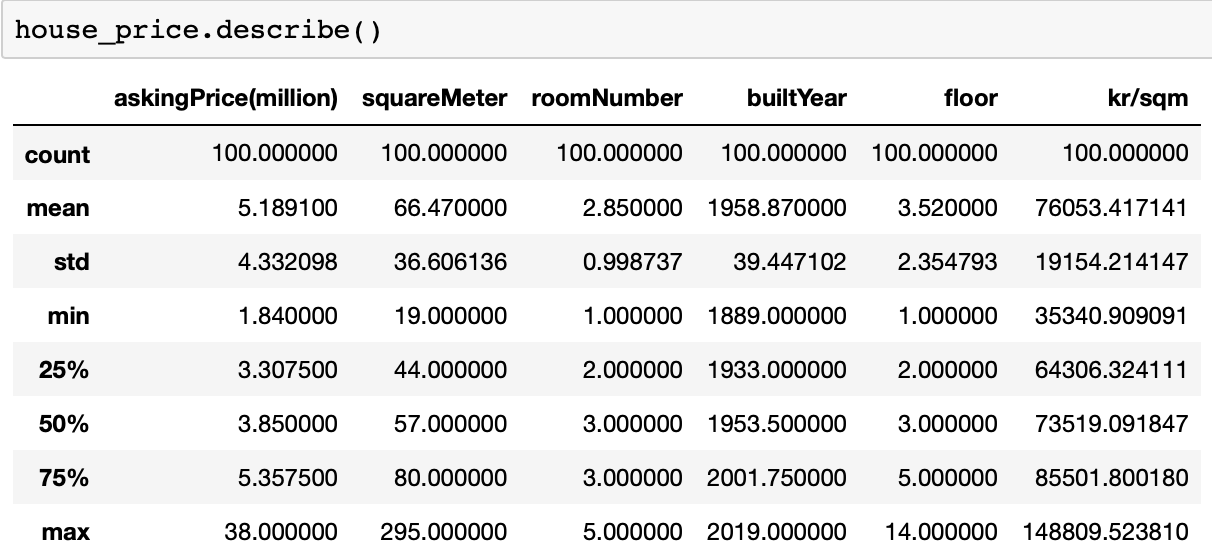
## 3.1 Distribution of number of rooms

*figure 3.1*

This bar chart shows the distribution of the number of rooms, from the apartments for sale in Oslo in December 2018. The most popular ones are two rooms apartment, just over three rooms apartments. These two observations equal approximately 65% of the hundred apartments for sale. The bar chart indicates that there are approximately twice as many three rooms apartments for sale than four rooms. In addition, we can see that the smallest occurrences are one and five rooms. Results are not very representative due to few observations. If the observations had been more, this would have been more representative for Oslo.

## 3.2 Analytical observations

In order to use python to calculate key figures and analytical observations we used the following function (figure 3.2). By using this function, we will get an overview of each variable. Some numbers we found interesting was that the average(“mean”) asking price in the data sample is 5.189100 million NOK. Whether or not a sample on 100 apartments gives you the correct average price for all apartments in Oslo is probably unlikely. In figure 3.2 you can also see the cheapest and the most expensive apartment.

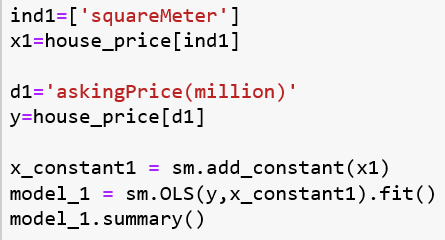
*figure 3.2*

## 3.3 House price analysis

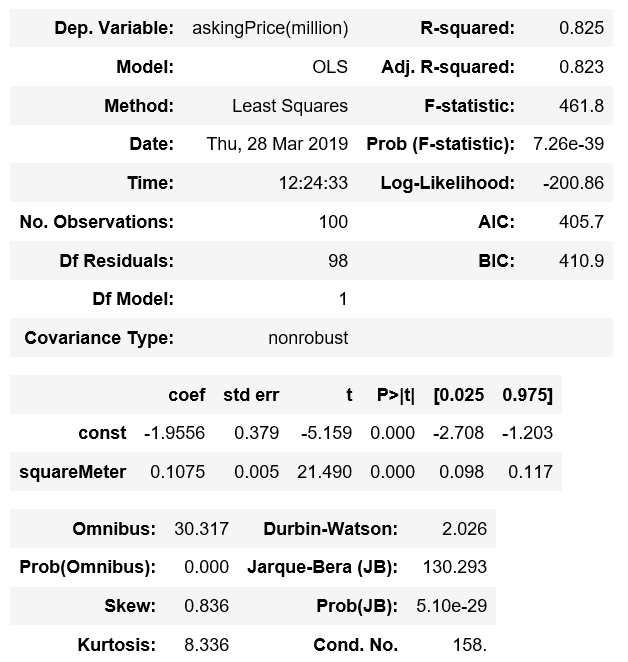
In task 4 we were given the task to perform simple linear regression. Linear regression is a statistic method used to explain the relationship between one dependent variable and one or more independent variables. You can also predict values of the dependent variable by putting actual values into the regression equation. By doing so you will receive a predicted value that estimates e.g. the cost of a 50-square meter apartment in Oslo. In addition, regression is used is to identify the strength of the effect that the independent variable(s) have on the dependent variable. For example, how much the price of an apartment increases or decreases when the number of rooms changes.

In task 4 we only used one independent variable because our task was to do simple linear regression.

Furthermore, we set “askingPrice(million)” as the dependent variable and “squareMeter” as the independent variable. In order to view the Statsmodel regression result we plotted the codes shown in figure 3.3:

Figure 3.3

The output:

Figure 3.4

### 3.3.1 Coefficient

The first thing we can look at is the coefficient for squareMeter. The coefficient shows us how much the dependent variable, “askingPrice”, changes when the independent variable, “squareMeter”, changes with one unit. In this case, we can tell that for each square meter the price of the apartments increases with 0.1075 million or approximately 100 000 NOK in average. This is as expected, and it is common that the bigger apartment the higher price.

The regression result gives us this regression equation:

**y = -1.9556 + 0.1075x**

### 3.3.2 Estimation

By inserting in different values for squaremeter (x) we can predict the result of askingprice (y) and compare different result with each other. We can also compare the predicted values with the actual values and see how good the regression equation is “fitted”.

The predicted price for a 50 and a 100 square meter big apartment in Oslo:

**y = -1.9556 + 0.1075 \* 50 = 3.4194 (million)**

**y = -1.9556 + 0.1075 \* 100 = 8.7944 (million)**

According to our calculated values, an apartment that is twice as large is about twice as expensive than the other apartment. That means that square meter and asking price does not increase linear, given the regression equation. If you compare the actual price of a 50-square meter apartment with the predicted one it miscalculates with 0.69 million (4.11 – 3.42). 0.69 million is the difference between the actual value and the estimated value. This is called residual.

If you are going to buy or sell a 50-square meter apartment and uses the regression equation above, it would be quite inaccurate. It depends on your view, but we think a difference of over half a million NOK would be quite vital both for the buyer and the seller Therefore, it can be problematic to use the regression equation if you are going to buy a predetermined size apartment and calculate the price by just by using the regression equation. You will just get an indication of the price, but you cannot determine if you. have enough money to buy that specific 50 square apartments in Oslo. More observations would give a more accurate prediction.

### 3.3.3 R-square

The difference between the actual value and the predicted one has to do with the R-squared value. R-squared tells you how much of the dependent variable that is explained by the independent variable. In this regression model R-squared is 0.825, which means that 82.5% of the variation in the price of an apartment is explained by the size of the apartment (square meter). On the other hand,17.5% in asking price is not explained. By adding more independent variables and by performing multiple regression, you can increase the R-squared. We will go through multiple regression later in this report.

### 3.3.4 Prediction vs. reality

Although the size explains much of the variation in the price of an apartment. It is relatively natural that variables like number of rooms, built year, which floor and the location affects the price of an apartment. It is commonly known that apartments located at St.Hanshaugen and Frogner are more expensive than apartments located at Oppsal and Mortensrud due to demand. Further on we are able to compare a 100-square meter apartment with a predicted 100-square meter apartment.

The residual between the estimated price of a 100-square meter apartment and an actual 100-square meter apartment:  8.79 – 7.50 = 1.29 million.

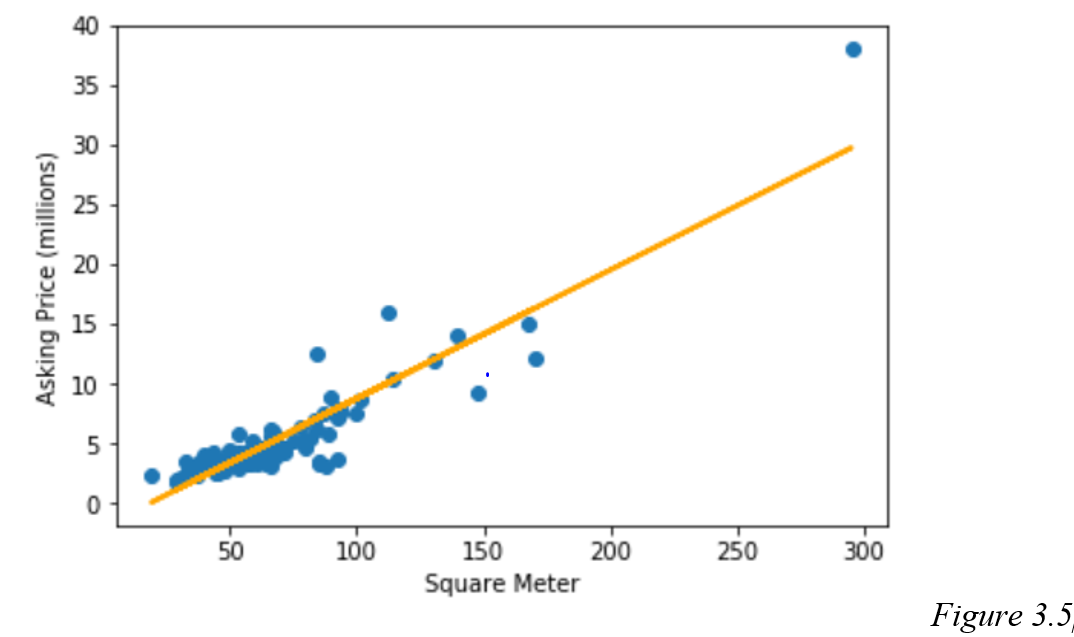
In addition, we can compare the residual from 100-square meter apartment with the residual we calculated from the 50-square meter apartment.

The residual between the estimated price of a 50-square meter apartment and an actual 50-square meter apartment:  4.11 – 3.42 = 0.69 million.

By analyzing these two results, we can conclude that the regression equation is more suited to estimate the price of smaller apartments than larger apartments. Because the residual is almost doubled when the size is doubled, and the regression equation is more accurate when estimating smaller apartments rather than bigger. Furthermore, it can be explained that the residual is larger on higher price apartments because we have more observations on lower price apartments. Despite the estimations we should be critical, because we only looked at one independent variable, and only looked at two different values of square meter and the effect it has on the price.

### 3.3.5 Trend line

Although we only looked at two different values in the examples above, we get another indication of our conclusion when we plotted the regression line and the actual values into a x and y label shown in figure 3.5. Just by looking at the actual values presented by the blue dots in figure 3.5, you can see that the values are closer to the linear regression line when the apartments are smaller, and the spread from the line is increasing when square meter is increasing. This supports our conclusion in our previous paragraph.



We can also set e.g. built year as the independent variable and compare it with the results we got above.

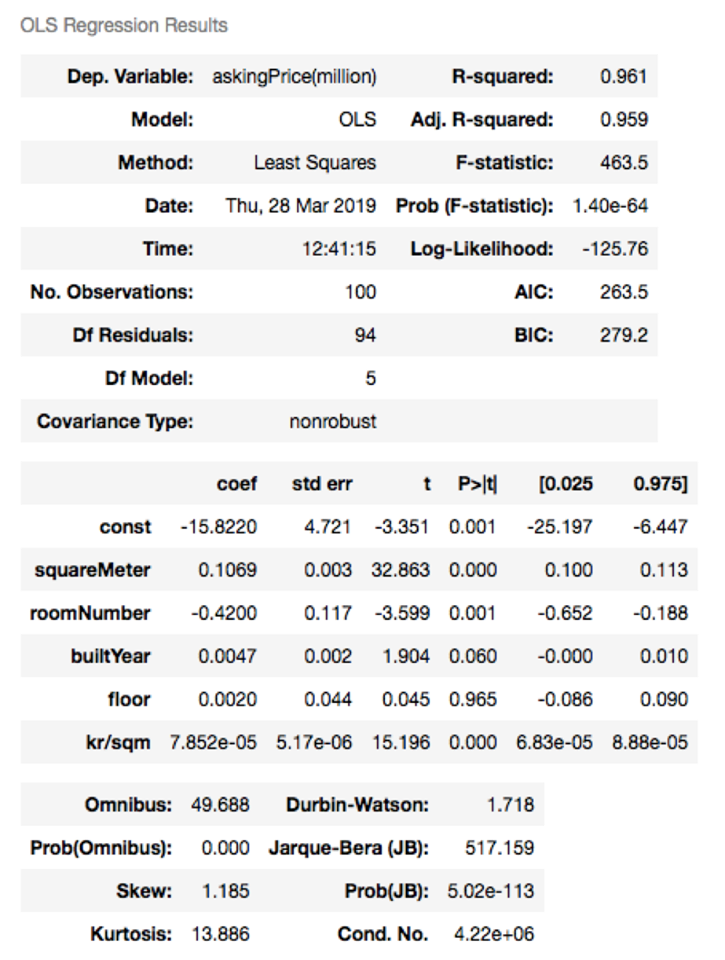
R-squared with built year as the independent variable is 0.016 or 1.6% which means that the built year has very little impact on the variation in asking price. This is also very intuitively because there are a lot of old apartment buildings in Oslo that have been renovated and other factors like the size of the apartment that affects the price much more than when it is built.

### 3.3.6 Correlation

The correlation coefficient with built year as the independent variable is 0.0141 which means that for each year built year increases the asking price increases with 14 100kr. The newer apartment the more expensive. If you compare the difference in price for an apartment built in 1900 and 2000 you would get a difference of 1.41 million (14 100 \* 100) when you predict the price difference. That equals approximately the price of a 13.5-square meter bigger apartment (1.41/0.1075 = 13.5). In other words, if you buy an apartment that was built in 1900 you would get a 13.5-square meter bigger apartment for the same price compared with buying an apartment built in 2000. You can also look at it the other way around and tell that for every 10th year older apartment you buy you would get approximately 1.35 square meter extra for the same price.

## 3.4 Multiple analysis

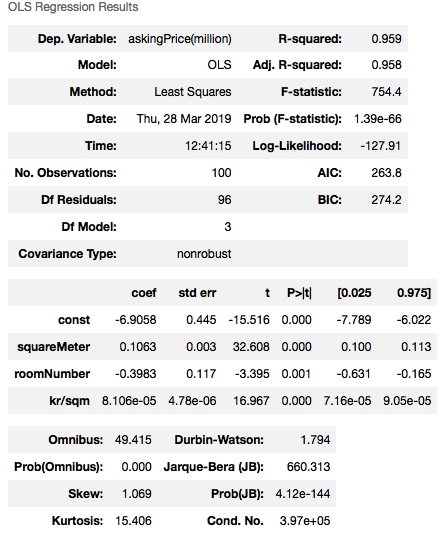
If we look at R-squared we see that it is close to perfection with 0.961, obviously 1 is a perfect match with all our scattered plots. But we know that R-squared will always increase the more variables you put in, therefore we have adjusted R-squared. Adjusted R-Square will compensate when adding more variables to give us a more accurate value, in this case: 0.959, not really a significant difference.



Next, we need to look at the coefficients for the constant and the independent variables. The constant is negative -15.822. This is not how it is in reality. House prices are usually not negatively priced. But in this case, it is necessary for the function to be correct. By looking at the coefficients of the independent values, we see that the amount of room numbers affects the dependent variable, house price, negatively. This does not necessarily make sense and might be the reason why the std error is so high with 0.117. Further on the function can behave like a concave feature. Initially, the offer rises to one specific point, where it then goes down. Therefore, this may not be so easy to predict.

If we include more observations, the std.error will decrease. On the other hand, the p-value indicates that we should include it in this analysis.

Built year has a significant impact on the asking price with 0.0047. However, if you look at built year and floors p-value, it is not significant with 0.06 and 0.965. Generally, if the p-value is under 0.05 you keep it, and over 0.05 you exclude it, when you use a 0.05 significance value. A p-value over 0.05 indicates that it is more than 0.05 or 5% probability that there is no connection between built year, floor and asking price. Therefore, we remove them in our new multiple linear regression model with Statsmodel:



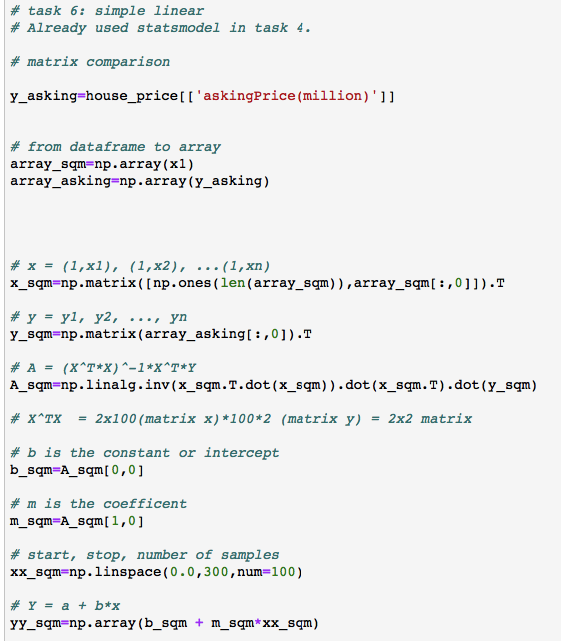
Adjusted R-Squared is nearly identical with the previous adjusted R-Squared, which is because of the non-significant independent variables ‘built year’ and ‘floor’. If you then add the x-values to the formula you get for apartment number 1:

**-6.9058 + 0.1063\*45 – 0.3983\*2 + 0.00008102\*59111= 1.87**

which is quite far away from the actual price of 2.66 million.

# 4 Simple linear regression coding

## 4.1 Using matrix



**Numpy.ones():**

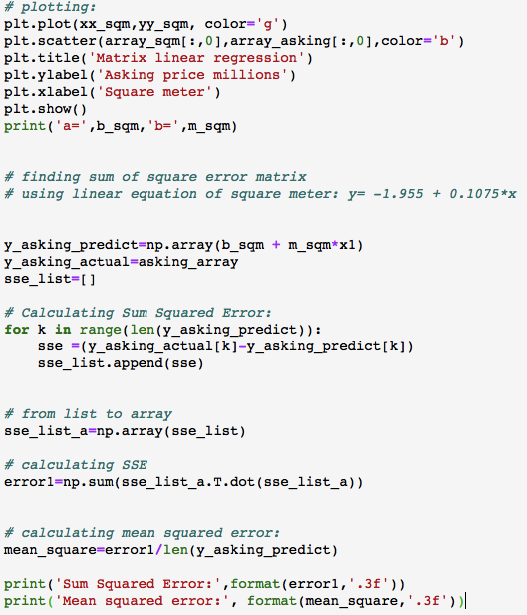
Before we begin solving A, we need to add ones in front of each x value in our matrix. This is to give us the intercept (b) when you take X and multiply it.

by A which contains b (intercept) and m (coefficient). Our matrix X then becomes a 100x2 matrix

**Numpy.linalg.inv.():**

To perform the inverse of A: A-1

**(.T)** is to transpose e.g. an array



**For loop:**

a loop so that we can append to our list. **Range()** and **len()** to determine when the loop should stop.

**Numpy.array():**

Converting to an array for mathematical purposes

## 4.2 Using Sklearn

**Lm.linearRegression():**

Creating linear regression object.

**.fit():**

there are four parameters here, where you can decide to e.g. calculate without the intercept, but at default it is true (includes intercept).

**.coef\_:**

coefficient

**.intercept\_:**

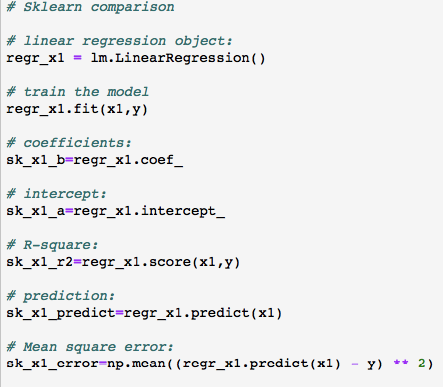
intercept

**.score():**

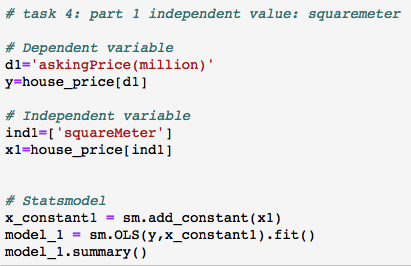
R-squared

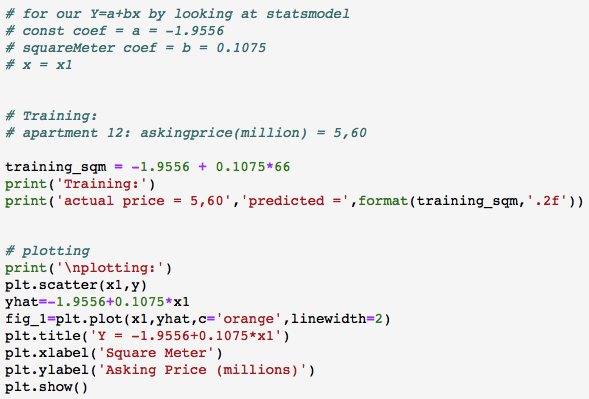
**.predict():**

prediction



## 4.3 Using Statsmodel

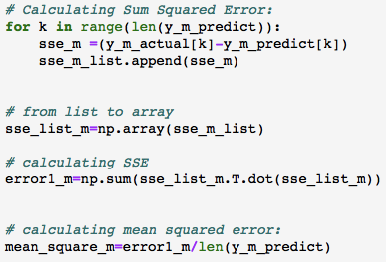




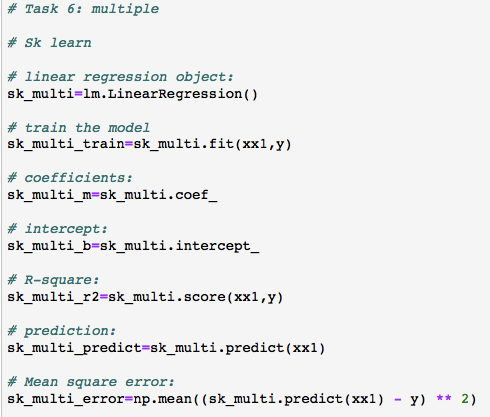
# 5 Multiple Regression coding

## 5.1 Using matrix

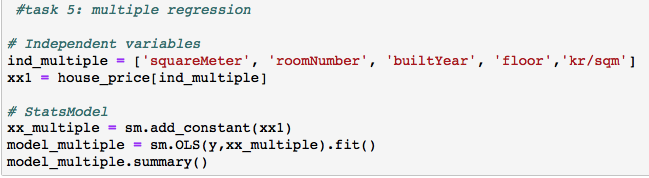




## 5.2 Using Sklearn



## 5.3 Using Statsmodel



# 6 Comparison

## 6.1 Matrix vs. Sklearn vs. Statsmodel

### 6.1.1 Matrix

What you receive is the equation Y = a + bx, a straight line where you can input different x values to see what effect it has on the dependent value. You can additionally calculate the sum of squared errors SSE= E^T\*E. However, there is a drawback using matrix. You will not receive R-squared, which is important to see the explanatory power of the model.

### 6.1.2 Sklearn

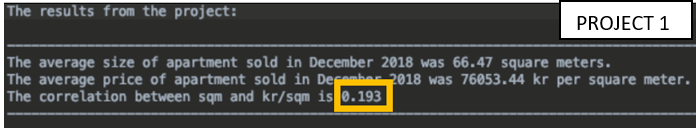
Sklearn gives you a broader option when it comes to key figures while doing a linear regression, compared to a matrix. We receive the R-squared, and we are then able to use a prediction function for the outcome. In the same way it is easier to extract and use the numbers you receive in Sklearn by making them into your own variables, like what we did to find the mean squared error by combining both Sklearn and the Numpy package.

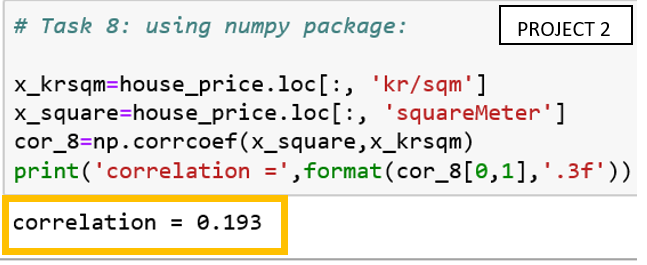
### 6.1.3 Statsmodel

By using Statsmodel we receive most of the key figures, and more importantly, Statsmodel provides adjusted R-squared compared to Sklearn and matrix. The biggest difference is the knowledge of the p-value of the different independent values coefficients, to see which one we might have to exclude. The downsides are that you must write the numbers manually, you cannot use some form of index to extract the numbers out, and you do not receive the total squared error nor the mean of it.

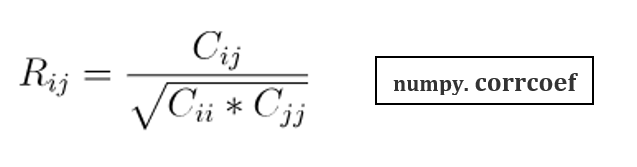
## 6.2 Project-1 and Project-2

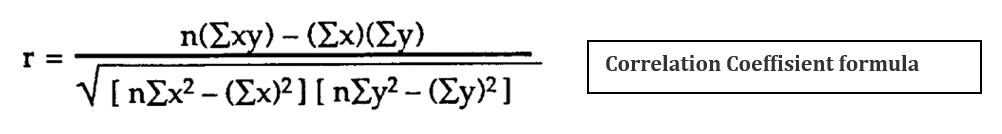
A correlation coefficient is a number between −1 and 1 which summarizes the degree of compliance. It is a statistical measure of how much two measurable quantities are related to each other. A positive number means that there is a positive correlation, and a negative number means that there is a negative correlation.





The correlation is positive 0.193 in both projects, showing that Numpy uses the same equation to calculate coefficient.





Our conclusion is that we can calculate the correlation coefficient manually or through built in functions in Python like Numpy. We receive the same results.

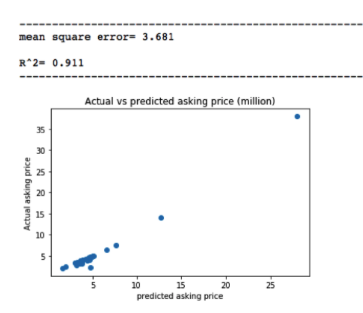
# 7 Machine learning

**NB! Every time you run the program the outcome of the prediction, R-squared and mean square error will change because it randomly takes out different data to the training set. All our calculations are based on the numbers we received while writing the report.**

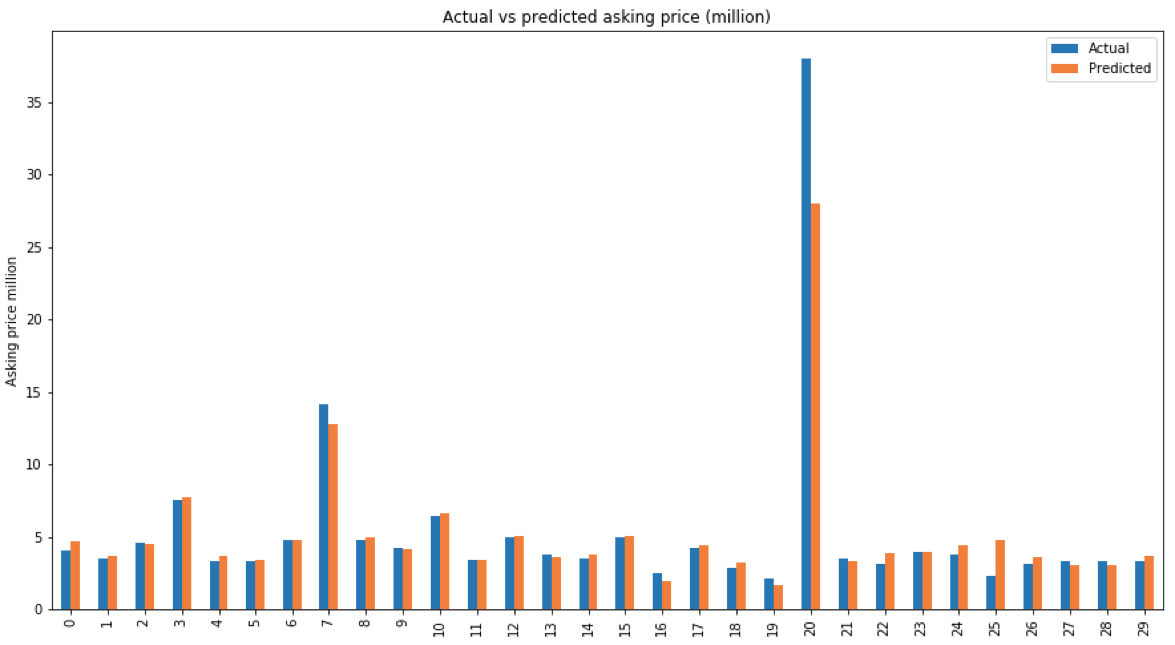
## 7.1 Testing and Training 30%

To perform Machine Learning we use the Sklearn package “train\_test\_split” and name it as “tts”. For the first comparison of actual and predicted prices, we test 30%, which means that 70 % goes to training and it predicts 30 %. To perform the test, we use the function train\_test\_split (tts), the first argument is our independent variables: “squareMeter”, “roomNumber”, “builtYear”, “floor”, kr/sqm. The second argument is the dependent value: “askingprice(million)”. The third argument is the test size which we set to 0.3 for 30%. That is basically the only difference between regular use of Sklearn and using machine learning. We receive the prediction, R-squared and mean square error.

Looking at R-squared and mean square error, we see that the prediction is quite accurate with R-Squared = 0.911. This means that the multiple linear regression explains 91.1% of the dependent variable, while the remaining 8.9% is explained by unknown external variables. Moreover, the mean square error is 3.681. This indicates a miscalculation at an average of 3.681 million, which may sound odd if the R-squared is 0.911. This might be explained by the 38 million NOK apartment that is in the mix of the predicted values as seen below.



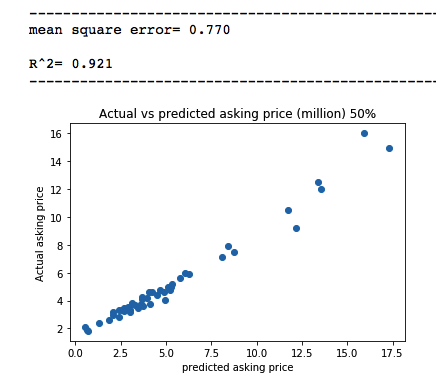
By plotting the values into a bar chart, it is easier to visualize it better, and easier see the difference.

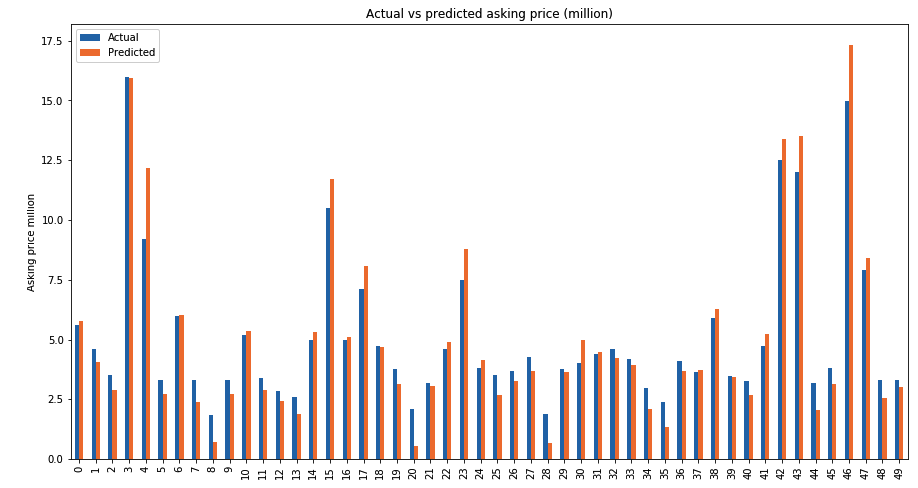


7.2 Testing and Training 50%

Same approach as before in our training/testing, but different split percentage.

This time with 0.5 instead of 0.3.





This one has a higher R-squared with 0.921, but this is also because R-Squared always will increase the more variables you have/include, so we should really compare the adjusted R-squared. Moving on to the mean square error, we can see that it is very low with 0.77, which is better compared to the previous result. One of the main reasons is that most of the predicted values are on the same price range, therefore it makes the prediction more accurate.