|  |  |
| --- | --- |
| 134,283 Amsterdam Stock Photos, Pictures & Royalty-Free ...  Amsterdam Housing prices  Business case | Abstract  In this business case, we will try to make predictive models to predict the housing price of a home in Amsterdam.  Rens van Dijck (608554)  Individual assignment |

Contents

[1 Introduction 2](#_Toc124696857)

[2 Objective 2](#_Toc124696858)

[3 Business understanding 2](#_Toc124696859)

[4 Data understanding 3](#_Toc124696860)

[5 Data preparation 4](#_Toc124696861)

[6 Visualisations 5](#_Toc124696862)

[6.1 Scatterplot Price Area 5](#_Toc124696863)

[6.2 Leaflets 5](#_Toc124696864)

[6.3 Cross variables 6](#_Toc124696865)

[7 Making the model 7](#_Toc124696866)

[8 Evaluating the model 8](#_Toc124696867)

[8.1 Prediction 9](#_Toc124696868)

[9 conclusion 9](#_Toc124696869)

[10 References 10](#_Toc124696870)

# Introduction

When thinking about a business case to use my skills learned in this Data Driven Decision Making minor for. I thought about looking at housing prices in Arnhem because that is where I currently live. After searching for data regarding Arnhem housing prices nothing usable came up. That is when I settled on doing the same project but for Amsterdam. Housing prices in Amsterdam are high so I was curious to inspect and use a data set I found online. This data set contained some interesting data to try and make a machine learning algorithm predicting housing prices and some useful visualisations.

The GitHub link: <https://github.com/RensvDijck/Amsterdam-Housing-Prices>

# Objective

The objective for this business case is make a machine learning algorithm that with the data available can make a prediction on housing prices in Amsterdam. While also providing useful visualisations of the housing prices in Amsterdam. This project will be made in R. The Crisp-DM process model is used to direct the steps in the project.

# Business understanding

The purpose of this project is to develop a machine learning algorithm in R for predicting housing prices in Amsterdam. The algorithm will be based on data collected from an online data set. The goal is to train and improve in the skills learned in the Data Driven Decision Making minor. This can also provide a tool for homebuyers to better understand the housing market in Amsterdam and make more informed decisions.

To achieve this goal, the project will involve several steps, including data cleaning, data manipulation, data analysis, and visualization. The data cleaning process will involve identifying and removing missing or irrelevant data, as well as dealing with outliers. The data manipulation process will involve transforming the data into a format that can be used by the machine learning algorithm, as well as creating new features or variables that may be useful for the analysis. The data analysis will involve exploring the relationships between different variables and identifying patterns or trends in the data. The data analysis will explore relationships between the different variables by using visualisations. The visualisations will contain charts and leaflet maps to give a clear and visual insight of the data.

The final product of this project will be a machine learning algorithm in R that can predict housing prices in Amsterdam based on the data collected and analysed. The algorithm will be evaluated using various performance metrics. The algorithm will also be tested by splitting the data into train and test sets.

# Data understanding

The data found online contained data about house prices in Amsterdam, with more information about each house. All the columns in the data set are as follows:

* **X:** The primary key used in the data set.
* **Address:** The address of the house, this contained the street name, street number and the city (Amsterdam) as well.
* **Zip:** This is the zip code of the house; this is not used because the longitude and the latitude were also available.
* **Price:** The price of the house in euro, this variable is used as the dependant variable in the models.
* **Area:** The area in square meters. This is interesting because you would expect that bigger houses are more expensive.
* **Room:** The number of rooms present in the house.
* **Lon:** The longitude of the house. Having the exact location of the house is interesting and can make some nice visualisations.
* **Lat:** The latitude of the house.

In the AmsterdamHuisPrijzen.qmd file the functions str(), summary() and head() are used to explore and understand the data, as well as inspecting the data set visually.

1. str(housing\_df)
2. summary(housing\_df)
3. head(housing\_df)
4. sum(is.na(housing\_df$Price)

The data set has 924 rows of house price data in Amsterdam, consisting of 8 columns. Also, to make sure there is no missing data we look for it. There seems to be some missing data in the Price column.

While exploring different machine learning algorithms I first tried to make a KNN model. After this model appeared to not be accurate at all, a linear regression model was used next to get a more accurate model.

A correlation plot was made of the data used in the KNN and linear regression model. The correlation between price and area seems a bit high. The correlation between area and room is also a bit high but these can both be logically explained.

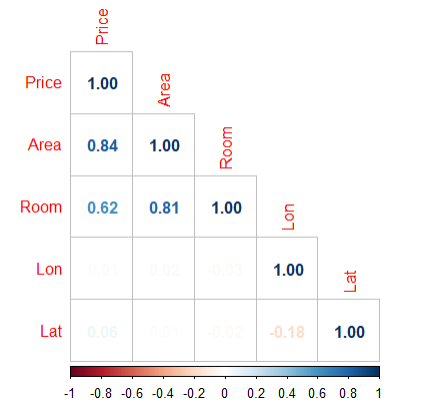


Figure - correlation plot

# Data preparation

To prepare the data before using it in the visualisations and the models, the missing values need to be removed. Then I check if there are still any NA values remaining to be sure. The primary key also is not used at all in the project, so it is removed as well.

1. clean\_df <- na.omit(housing\_df)
2. sum(is.na(clean\_df))
3. clean\_df <- clean\_df[,-1]

For the models some more cleaning is necessary, so we do some more preparing the data for that. The Address and Zip column are removed because they are not used in either the KNN or the linear regression model.

1. model\_df <- clean\_df[, -c(1:2)]

While trying to make the KNN model and testing for accuracy of the models some code was used to remove outliers and scale the data. These methods are not used in the models because it decreased the accuracy of the model. The outliers are natural, so they are not removed. And the independent variables are close enough to each other to fit in the models.

1. model\_df <- filter(model\_df, Price < quantile(Price, 0.98))
2. model\_df <- filter(model\_df, Area < quantile(Area, 0.98))
3. model\_df[, c("Area", "Room", "Lon", "Lat")] <- scale(model\_df[, c("Area", "Room", "Lon", "Lat")])
4. head(model\_df)

# Visualisations

The visualisations presented in the project can be viewed in the project file. In this document I am going to explain what each visualisation presents. And which conclusions can be drawn from them.

## Scatterplot Price Area

The first graph presented in the project is a scatter plot with on the Y-axis the Price of the houses, on the X-axis the area in m2, and the hue of the data points is dependent on the number of rooms in the house.

1. ggplot(clean\_df, aes(x = Area, y = Price, color = Room)) +
2. geom\_point()

The second graph is the same but with a reduced max price, area, and rooms, to give a better visual and the spread of most of the data. Price max = 2.000.000, area max = 250, room max = 6.

1. ggplot(clean\_df, aes(x = Area, y = Price, color = Room)) +
2. geom\_point() + scale\_y\_continuous(limits = c(0, 2000000)) + scale\_x\_continuous(limits = c(0, 250)) + scale\_color\_continuous(limits = c(0,6))

What can be derived from this is this graph seems quite linear and that the most expensive houses in Amsterdam have a lot of area and rooms.

## Leaflets

The second graphs that were made used the longitude and latitude to visualise the map of Amsterdam using a R package called leaflet. Also, the price is used to color the data point in a gradient. The color palette used ‘YlOrRd’ gives a gradient from pale yellow up to dark red. The first leaflet doesn't really give a lot of information with the color gradient but does give all the data points a visual location. Also, a label is added to each data point giving the Address and Price of the house.

1. leaflet\_df <- clean\_df
2. colorLeaflet <- colorNumeric(palette = "YlOrRd", leaflet\_df$Price)
3. label <- paste("<strong>", leaflet\_df$Address, "</strong><br>Price: ", leaflet\_df$Price) %>% lapply(htmltools::HTML)
5. leaflet(leaflet\_df) %>%
6. addTiles() %>%
7. addCircles(~Lon, ~Lat,
8. color = colorLeaflet(leaflet\_df$Price),
9. opacity = 1,
10. label = ~label
11. )
12. leaflet\_df <- subset(clean\_df, Price < 2000000)
13. colorLeaflet <- colorNumeric(palette = "YlOrRd", leaflet\_df$Price)
14. label <- paste("<strong>", leaflet\_df$Address, "</strong><br>Price: ", leaflet\_df$Price) %>% lapply(htmltools::HTML)
16. leaflet(leaflet\_df) %>%
17. addTiles() %>%
18. addCircles(~Lon, ~Lat,
19. color = colorLeaflet(leaflet\_df$Price),
20. opacity = 1,
21. label = ~label
22. )

In the second leaflet, the price is maxed out at 2.000.000 to give a better view of the color gradient at work. Now it is way easier to see which houses stand out among the rest.

## Cross variables

In this part I set up every pair of notable variables up against each other with one on the X-axis and one on the Y-axis. This gives insight in the relationships between variables.

1. ggplot(clean\_df, aes(x = Lon, y = Price)) + geom\_point() + ggtitle("Price vs Longitude")
2. ggplot(clean\_df, aes(x = Lat, y = Price)) + geom\_point() + ggtitle("Price vs Latitude")
3. ggplot(clean\_df, aes(x = Room, y = Price)) + geom\_point() + ggtitle("Price vs Rooms")
4. ggplot(clean\_df, aes(x = Area, y = Price)) + geom\_point() + ggtitle("Price vs Area")
5. ggplot(clean\_df, aes(x = Area, y = Room)) + geom\_point() + ggtitle("Area vs Room")
6. ggplot(clean\_df, aes(x = Lon, y = Area)) + geom\_point() + ggtitle("Longitude vs Area")
7. ggplot(clean\_df, aes(x = Lat, y = Area)) + geom\_point() + ggtitle("Latitude vs Area")

Some interesting observations here include that in Price vs Longitude and Price vs Latitude, it is very clear that in the centre of both the longitude and the latitude the Price is higher with more outliers. This means that in the centre of Amsterdam houses are more expensive on average. And the most expensive houses are also located in the centre. The same can be said for Longitude and Latitude vs Area, here in the centre of both longitude and latitude most of the outliers lie as well.

# Making the model

At first the plan was to make a KNN model with the data available, however this model was not accurate at all so a new plan was devised to make a linear regression model because both the variables area and room appeared to be linear. The code trying to make the KNN I left in because it is part of the process of coming to a better model. In the KNN model a k of 32 was used for the best accuracy and was evaluated with use of a confusion matrix.

To start making the model, the data has to be split in a training set and a testing set. This can be done with the createDataPartition function from the caret package. The source data model\_df now has only the variables: Price, Area, Room, Lon, Lat.

1. set.seed(123)
2. msplit <- createDataPartition(model\_df[, "Price"], p = 0.75, list = FALSE)
3. train\_df <- model\_df[msplit,]
4. test\_df <- model\_df[-msplit,]

Now with the training set we are going to train the linear regression model. No transforming of the data was done before because it decreased the accuracy of the model. Also removing outliers is not done because in this case, the outliers are natural.

1. model\_lm <- lm (Price ~ Area + Room + Lon + Lat, data = train\_df)
2. summary(model\_lm)

# Evaluating the model

To evaluate the linear regression model, we look at multiple performance metrics like adjusted R squared, the RMSE and variables that are statistically significant.

Table - linear regression model, Amsterdam housing

|  |  |
| --- | --- |
|  | Housing price |
| Intercept | -45924193.84 |
|  | (24956761.84) |
| Area | 8663.09 \*\*\* |
|  | (316.03) |
| Room | -36477.14 \*\* |
|  | (11800.49) |
| Lon | -170720.47 |
|  | (210180.64) |
| Lat | 891511.90 |
|  | (472444.90) |
| R^2 | 0.72 |
| Adj. R^2 | 0.72 |
| Num. obs. | 692 |

\* = p < 0.05,

\*\* = p < 0.01,

\*\*\* = p < 0.005

Standard error between brackets. All continuous variables standardized.

Above is the regression table of the linear regression model of housing in Amsterdam. Using price as the dependent variable, and area, room, lon and lat as independent variables. The adjusted R squared is 0.72 which means that 72% of all prices can be described with the independent variables. The RMSE of the prediction vs the test set is 316537.8.

Area:   
For the variable Area, it has a high significant statistical significance on price. This can be concluded because the p value is under 0.005. It seems that area per square meter is a big deciding factor on the price for houses in Amsterdam.

Room:  
The variable Room also has some statistical significance but not as much as Area since the p value is 0.002. This seems logical since more expensive houses typically have more rooms, but this is not a characteristic of expensive houses.

## Prediction

Before making the models, the data set is split in two sets. One training set with 75% of the data and one test set with 25% of the data. Now we are going to use the test set to test how the predicted prices hold up against the test set actual prices.

1. predict <- predict(model\_lm, newdata = test\_df)
2. results <- data.frame(actual = test\_df$Price, predicted = predict)
3. ggplot(results, aes(x = actual, y = predicted)) +
4. geom\_point() +
5. geom\_abline()

A plot is made in ggplot to set the actual values of the test set to the x-axis and the predicted values on the y-axis. In this graph there is also a line in the middle to see how close the prediction is vs the actual data.

In this graph it is made clear that the accuracy of the model is better at lower prices since it deviates less from the line at the smaller price ranges. When going past homes of more than 1.000.000 euro the accuracy decreases. There are some outliers in this model, but they are deemed natural since the prices of the houses are there for a reason. Just with the data available that cannot be shown.

# conclusion

As the conclusion of this project. The linear regression model made during this individual project can be useful in predicting housing prices. With an adjusted R squared of 0.72 it is accurate enough to give predictions, but it can still be improved a lot. The visualisations made can be very useful, a leaflet of Amsterdam is easy to understand and can give a lot of information for homebuyers.

However, the linear regression model is not perfect, a lot of other methods and techniques could be attempted to improve the model. Also, the data was not complete. With more data the model could have been more accurate.

In the end this project was about increasing and using the skills learned in the Data Driven Decision Making minor. This I think has been successful because it was a fun project to work and improve my skills on. Trying to make the KNN model has taught me a lot as well like how to evaluate and retest a model that I wasn’t that familiar with. Working in R has also improved my coding skills a lot I feel and has been a good experience for further in my professional career.

# References

Tilbe, A. (2022, September 27). *K-Nearest Neighbor (KNN): Why do we make it so difficult? simplified*. Medium. Retrieved January 15, 2023, from <https://levelup.gitconnected.com/k-nearest-neighbor-knn-why-do-we-make-it-so-difficult-simplified-701573a862b0>

Thomasnibb. (2021, August 24). *Amsterdam house price prediction*. Kaggle. Retrieved January 15, 2023, from <https://www.kaggle.com/datasets/thomasnibb/amsterdam-house-price-prediction?datasetId=1549462>

Hotz, N. (2023, January 8). *What is CRISP DM?* Data Science Process Alliance. Retrieved January 15, 2023, from <https://www.datascience-pm.com/crisp-dm-2/>

Kinha, Y. (n.d.). *An easy guide to help you decide which machine learning algorithm to ...* An easy guide to help you decide which Machine Learning algorithm to use for your business problem. Retrieved January 15, 2023, from <https://edvancer.in/an-easy-guide-to-help-you-choose-which-machine-learning-algorithm-to-choose-for-your-business-problem/>