

Quantitative Data Analysis – Exercises

(Week 07)

In these exercises, you will learn:

- to perform simple and multiple linear regression analyses.
- to perform regression analysis based on regression trees and random forests.

In the data analysis process model, these exercises cover part of the steps “Statistical data analysis and/or Modeling” and “Evaluation & Interpretation” (see figure 1). Results of the exercises must be uploaded as separate files (**no .zip files!**) by each student on Moodle. Details on how to submit the results can be found in the tasks below.

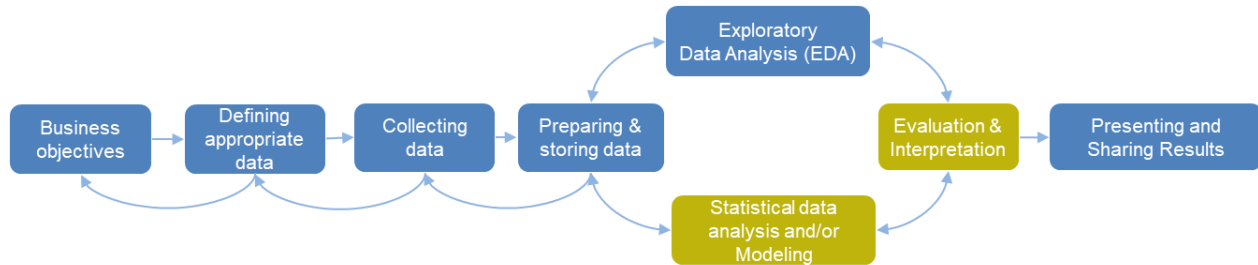


Figure 1: Data analysis process model (see slides of week 01)

Task 1

In this exercise, you will learn to perform simple and multiple linear regression analyses. The tasks are:

- Run the Jupyter notebook '[linear_regression.ipynb](#)' step by step and try to find out, what the Python code does.
- Go to the section 'Simple linear regression ...'. Create a new simple linear regression model with `price_per_m2` as target variable and `area` as the explanatory variable. Check the R-squared value of this new model. State whether the R-squared value is higher compared to the original model with `price` as target variable and `area` as the explanatory variable. Also check the histogram of model residuals. State whether the residuals are normally distributed or not.
- Create a new simple linear regression model with `price_per_m2` as target variable and `rooms` as the explanatory variable. Check the R-squared value. State whether the R-squared value is higher compared to the original model with `price` as target variable and `area` as the explanatory variable? Also check the histogram of model residuals. State whether the residuals are normally distributed or not.
- Go to the section 'Multiple linear regression ...'. Include the variables `tax_income` and `dist_supermarket` as additional variables in the model. State whether the variables are statistically significant (at the 5% significance level). Note that this

can be figured out by looking at the part of the output-table shown below. If the value $P > |t|$ is smaller than 0.05, than a variable is statistically significant (at the 5% significance level).

	coef	std err	t	$P > t $	[0.025	0.975]
const	458.4991	65.979	6.949	0.000	328.937	588.061
area	15.0355	0.639	23.543	0.000	13.781	16.290
pop_dens	0.2381	0.012	20.189	0.000	0.215	0.261

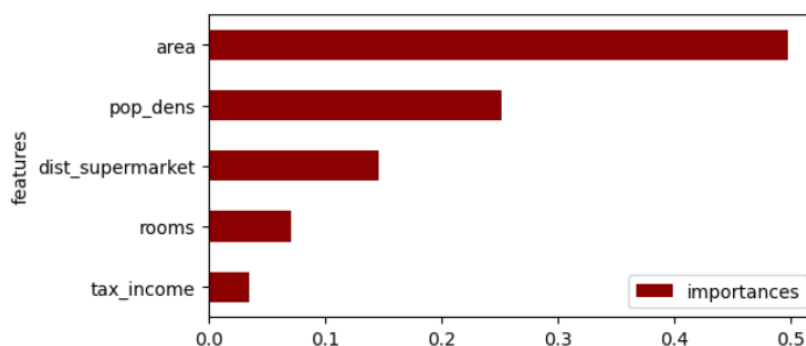
To be submitted on Moodle:

- The Jupyter notebook as html-file '[linear_regression.html](#)' with the changes and short explanations according to b), c) and d).

Task 2

In this exercise, you will learn to perform regression analyses based on a regression tree and a random forest. The tasks are:

- Run the Jupyter notebook '[regression_trees_random_forest.ipynb](#)' step by step and try to find out, what the Python code does.
- Go to the section 'Fit the regression tree model'. Change the model parameter `max_depth=3` to `max_depth=5`. This will change the depth (complexity) of the tree. Compare the output (text and graphic) with the output of the original regression tree (in which `max_depth=3`). Do you see any differences? In the Jupyter notebook, explain why.
- Go to the section 'Calculate coefficient of determination ...'. Look at the coefficient of determination (R-squared). Now, go to the section 'Create train and test samples for the regression tree ...'. Drop the two variables `area` and `rooms` from the train and test samples. Run the Jupyter notebook again. Does the R-squared value change? If so, explain why in the Jupyter notebook.
- Go to the section 'Show feature importance' of the random forest and look at the barchart. Note that 'features' is another name for the 'explanatory variables' in a Machine Learning (ML) model. The barchart should look like this:



- e) Go to the section 'Create train and test samples' and drop the variable area. Run the Jupyter notebook again.
- f) Does the importance of features change? If yes, explain why in the Jupyter notebook.

To be submitted on Moodle:

- The Jupyter notebook as html-file '[regression_trees_random_forest.ipynb](#)' with the changes and short explanations according to b), c), d), e) and f).