Machine Learning Capstone Proposal

Christoph Hilty

01. July 2018

1 Domain Background

The project is an ongoing kaggle competition[1] based on the 'Ames Housing Dataset'[2] of Dean De Cook. With a modernized and expanded data base the task provides an interesting use case for various data science and machine learning methods. The prediction of house prices is clearly a much needed and important part in the real estate world and there is a lot of money involved. An accurate prediction is of high importance for the seller and the buyer as well. With a lot of features and a supposedly easy problem, the task offers a lot of room for data analysis, feature engineering and a variety of machine learning algorithms. This makes the project in my opinion very attractive and engaging and allows me to collect my first experiences with a kaggle competition. I moreover think, that the simple nature of the project will allow me to focus more on the different methods and algorithms learned in the course.

2 Problem Statement

Assignment of the project is to predict the final price of different residential homes in Ames, Iowa based on a lot of different explanatory features provided in the dataset[3].

3 Datasets and Inputs

The data for the project[3] provides 79 explanatory variables describing almost every aspect of residental homes in Ames, Iowa and can be seen as an expanded version of the Boston Housing dataset. The data is provided by the competition.

4 Solution Statement

The first step will be to closely examine the features and their correlations. Different features engineering methods should be able to augment the data with valuable information. After the data cleansing and augmentation part different algorithms can be used, starting with linear regression and optimizing the root mean squared deviation:

$$RMSD = \sqrt{\frac{\sum_{t=1}^{T}(\hat{y}_t - y_t)^2}{T}}$$

This method can be combined with advanced techniques like ensembles and boosting.

5 Benchmark Model

For the benchmark model a comparison with the leaderboard[4] will be used. For this approach a histogram (cropped to submissions with a root mean squared error of below 2) is shown in the following figure:

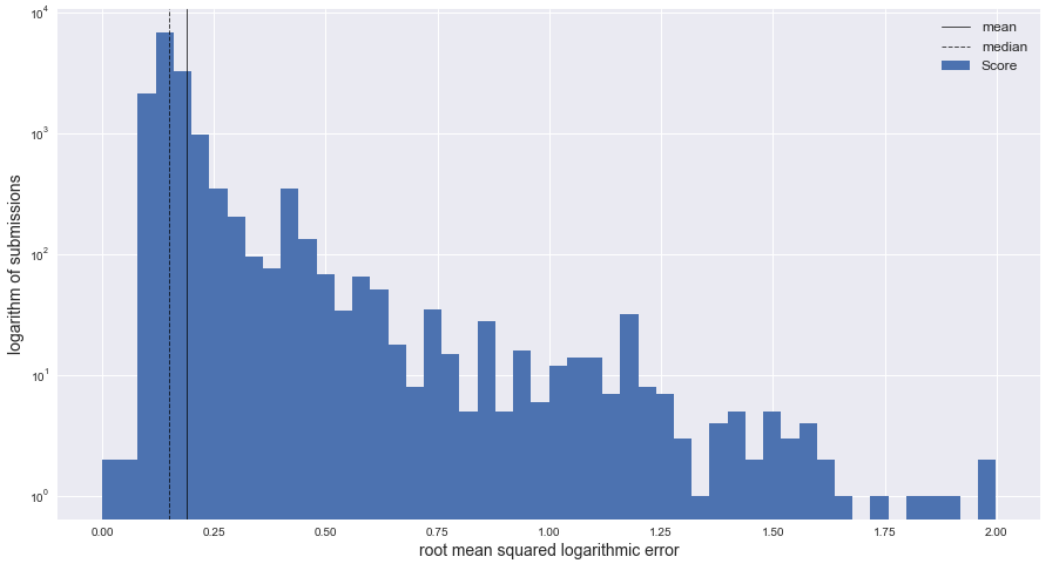

Figure 1: The leaderboard of the competition

6 Evaluation Metrics

The evaluation metric for the competition[1] is the root mean squared error between the logarithm of the predicted and the logarithm of the observed price. This leads to the following objective function:

$$J(\theta) = \sqrt{\frac{\sum_{t=1}^{T}(\log price_t^{(predicted)} - \log price_t^{(observed)})^2}{T}}$$

The logs are meaning, that the error in prices for cheap houses are punished equally as the error for expensive houses.

7 Project Design

The project will be a consisting of jupyter notebook in which the data is analyzed, the algorithms implemented and the appropriate visualizations are displayed in an appealing manner. In a first step some initial statistics for the different variables are displayed and possible correlations between the data are revealed and missing data is correctly addressed. An analysis of the principal components could further improve the understanding of the directions of the data. With this insight the engineering of the features is started and more concise and relevant features are created which do eliminate possible detrimental co-linearity. In a next step the data is prepared for the regression by encoding the variables in the correct formats. Normalization and standardization has to be considered for all the features and has to applied where necessary. Also categorical variables are divided into binary variables (One Hot Encoding). After these pre-processing steps, a simple regression is applied and some benchmarks are collected. Then in a next step more advanced regression methods are implemented (e.g. RandomForests, GradientBoosting e.g.) and the model gets incrementally sophisticated. Always keeping the optimization function in mind and questioning additional complexity.

For the parameter optimization of the different algorithms a grid search will be applied where suitable.

If time permits, other models (e.g. DecisionTreeRegressor[5]) could be evaluated. An interesting approach would be to further try a deep learning method with a neural network (using the keras wrapper object for the Scikit-Learn API[6] called KerasRegressor[7]) and compare the results to the regression techniques.

Because of the different steps in the project a data pipeline[8] may provide some benefits and will be considered for implementing.

References

[1] kaggle.com. House prices: Advanced regression techniques. https://www.kaggle.com/c/house-prices-advanced-regression-techniques.

[2] Dean De Cook. Alternative to the boston housing data as an end of semester regression project. https://ww2.amstat.org/publications/jse/v19n3/decock.pdf.

[3] kaggle.com. Dataset. https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data.

[4] kaggle.com. Leaderboard. https://www.kaggle.com/c/house-prices-advanced-regression-techniques/leaderboard.

[5] scikit learn.org. sklearn.tree.decisiontreeregressor. http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html.

[6] scikit learn.org. Api reference. http://scikit-learn.org/stable/modules/classes.html.

[7] keras.io. Wrappers for the scikit-learn api. https://keras.io/scikit-learn-api/#wrappers-for-the-scikit-learn-api.

[8] scikit learn.org. sklearn.pipeline.pipeline. http://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html#sklearn.pipeline.Pipeline.