# Comparison of Lineal Regression Vs Decision Tree Regression

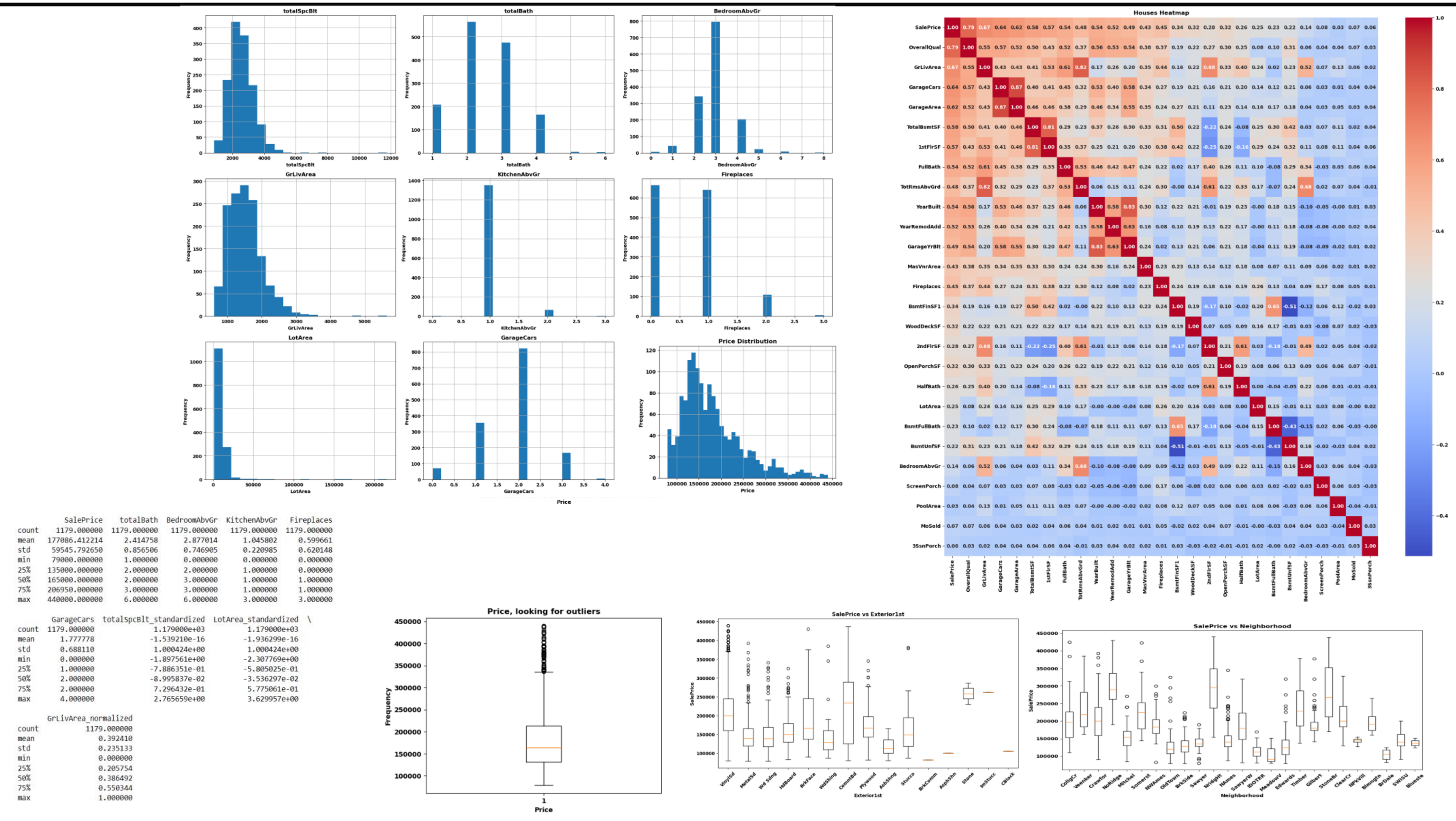
**Coursework:** INM431 Machine Learning **Name:** Antonio Jose Lopez Roldan **Email:** antonio-jose.lopez-roldan@city.ac.uk

## Description and Motivation

- Building and comparing two machine learning models: Lineal Regression vs Decision Tree Regression.
- This piece of work aims to clarify if houses prices can be accurately predicted by the most basic variables that every house should have such as kitchen, living room space, total space built, etc.
- The process followed to build the models was kept as simple as possible. When data is limited, simpler models are many times more effective [1].
- Common sense suggests that houses prices should be predictable based on the kind of the aforementioned features.
- Since I wanted to find my very own discoveries, only few references were briefly read before starting, and during the project. Most of the text relay on my own experience/analysis. There isn't many specific references, although used them as general guide in the whole process .

## Initial Analysis of Dataset

- The chosen data set for the project is the "Ames, Iowa House Market" compiled by Dean De Cock [2].
- The original data set is form by features that every house could have (pool, kitchen, living room, lot area, etc).
- The original data set has1460 row x 81 columns (Including "Id"). Some columns were deleted since they had many empty values. After data cleaning, 1179 records x 36 columns were left to do the analysis.
- There were many outliers across the different variables and as example we can take a look at the target variable: "SalePrice".
- Some of categorical variables seem to have a high variability. For instance, sale price-neighbourhood shows a significant variation in the mean of the house prices this means that the location definitely affect to the price. However, others also shown similar behaviour but after further examination the percentage of the houses affected by those variables in their price were quite small is the case of the sale price vs exterior1st (exterior covering the house).
- As was diving into the data set, the former motivation changed from "trying to build two models to make accurate predictions, then compare them" to focus on the basic features that every basic house should have; lot area, bathroom, garage, kitchen, and so on as well as others relevant such as neighbourhood. I understood that to build better models I should start by the basics.
- Applied feature engineering to some of the features(Standardization, normalization, one-hot encoding). Created new variables combining others. The feature totalSpcBlt_standardized is a standardization of the figures for the total space built (surface built in first floor + surface built second floor, etc).
- Looking at the correlation heatmap we can easily spot the highest correlated variables against the dependent variable "SalePrice". It is worthy to mention the variables GrLivArea (ground living area) coefficient of 0.67 or "GarageCars" with a coefficient of 0.64. Even though OverallQual (overall material and finish quality) has the strongest correlation with a coefficient of 0.79 has not been included since the focus of the study is "tangible house parts".
- Examined the data distribution to determine its compatibility with the models. Data distribution of the Sale Price, total space built, and ground living area is right-skewed whereas lot area follow a normal distribution.
- Multicollinearity was checked by applying VIF (Variance Inflation Factor), many columns had multicollinearity. High multicollinearity may make the model unstable therefore, some of the columns were deleted [3].




## Linear Regression

- Linear Regression is a model that assumes a relationship between the dependent variable and the independent variable.
- Linear Regression predict the dependent variable based on a function of the independent variables. It uses a linear equation Y = a + b*X where "a" is the intercept and "b" is the slope of the lineal representation.
- The coefficients measure the effect of each independent variable on the dependent var iable. They are denoted by the Greek letter β. The intercept is represented by β0 and indicates the value of "Y" when x = 0. Whereas β1 represents the slope, the slopes indicates how much changes "Y" for each unit of β1.

### Pros

- Simplicity: Linear regression is easy to grasp, therefore, is very appropriate for situations where transparency is crucial.
- Computational efficiency. The model require from little physical resources this means is an agile model even when training large datasets.
- Suitability for continuous data, this is, Lineal Regression performs well with continuous data and for our case (house prices) is an idyllic option.
- It has the potential to be highly accurate when the relationship between variables is linear.

### Cons

- Sensitivity to outliers. In this sense, W. Choi highlight the impact of outliers on linear regression models.
- Assumes linearity. This is definitely not the real-world scenario where sometimes there isn't any relationship between variables.
- Multicollinearity. The model takes for granted that there is not high correlation between the independent variables, these presumption are wrong and can lead to make unstable estimations of the model.
- There is a high risk of overfitting specially in cases where the model includes multiple independent variables.

## Decision Tree Regression

- Decision Tree is a non-parametric supervised algorithm, it means that they don't need a predetermined set of parameters to start the training.
- The model is similar to an upside-down tree. It consists of a root node that represent the entire population, decision nodes that split the data into smaller subsets until the terminal node. Leaf (or terminal) nodes represent a class label in the data according to the path taken.
- It tries to split the data into smaller groups based on the features, at the time also try to make as similar as possible in terms of the target value. The final prediction is based on the terminal nodes.

### Pros

- Easy to understand and interpret, the decisions can be seen through the diagram tree.
- Can handle both numerical and categorical data. Can address multi-output problems.
- They do automatic feature selection by identifying the best features for splitting data
- Can manage big amounts of data sets without the need to adjust the range of different features before using them.

### Cons

- Decision Trees are prone to overfitting, especially when the tree is deep. This fact can make the model sensitive to fluctuations.
- Small variations in the data can lead to the generations of different trees making them not that reliable for making predictions.
- Simple decision trees may not have the same capacity to make predictions as models are more complex this can lead to results with the presence of bias.
- Decision Trees have high variance and can create too many complex trees that don't fit well in new data.

## Hypothesis Statement

- Linear Regression model should perform better since it is supposed that the selected features have a strong relationship with the target variable, however in contrast with Decision Tree Regression, linear regression might not capture complex non-linear relationships therefore the performance of both models will probably be similar.
- Overall, we expect that the Linear Regression model will outperform the Decision Tree Regression model [4]
- Perhaps seems too obvious but errors in the prediction of house prices will likely be more significant in houses that deviate from the average price.

## Methodology

- Set a seed for reproducibility of the two models.
- "Holdout validation technique" was used. Data was split into 70% training, 30% test.
- Built a simple Linear Regression model considering all the original variables. Examining model performance gave me a general insight into the whole (models, features, etc).
- Based on my years of experience in the construction field, I proceeded to make a first selection of features, built another model with these features and examined the results.
- Feature selection. Applied Lasso regression for variable selection and regularization, it helped to identify and select the most relevant features for the final model.
- Built another linear regression model, examined the results.
- Continued with feature selection, applied VIF(Variable Inflation Factor) to check multicollinearity. Based on the results, deleted highly correlated features to avoid overfitting or bias in the models.
- Built different versions of the two models and checked results (R-squared, Root Mean Squared Error, Mean Absolute Error, Residuals, charts results for the two models). Selected final parameters such as " maximum number of splits".

## Choice Of Parameters And Experimental Results

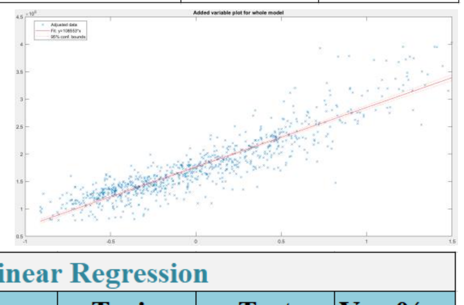
### Common points for both models

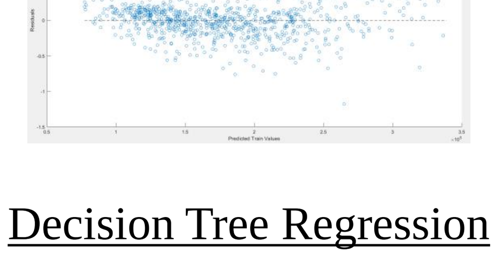
- Feature selection for both models was initially performed discretionarily, followed by the application of Lasso regression and the Variance Inflation Factor(VIF) analysis.
- Initial results before feature selection were unstable and performance was quite poor. The residual errors were over 100,000 units for example.
- Plot the residuals Vs predicted train values, and residuals vs predicted test values for further understanding of the metric. The charts for each model are below.
- Calculated the formulas for MAE, RMSE through coding in MATLAB.
- The biggest difference between models happens in the test. Right side upper corner.

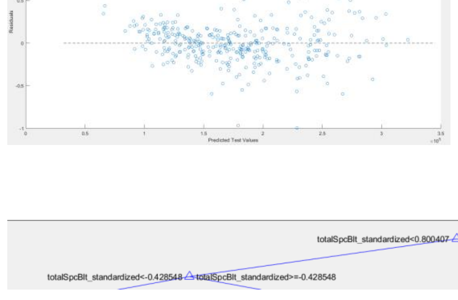
| Linear Regression Vs Decision Tree Regression | | |
|---|---|---|
| **Metric** | **Train** | **Test** |
| Mean Absolute Error | 12.37% | 20.68% |
| Root Mean Squared Error | 8.78% | 22.56% |
| R-squared | 3.25% | 11.96% |

### Linear Regression

- On the right side, the chart for the final Lineal Regression model before any evaluation against training/test models. this is, predictors Vs target value.
- Parameters were automatically defined by MATLAB. "fitlm" function estimates the parameters for each predictor by minimizing the sum of the squares of the residuals.



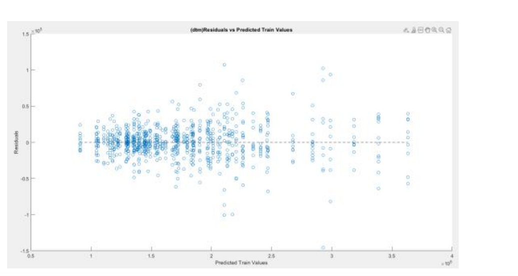


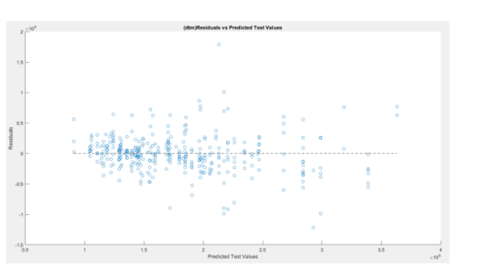
| Linear Regression | | | |
|---|---|---|---|
| **Metric** | **Train** | **Test** | **Var %** |
| Mean Absolute Error | 17676.76 | 17545.62 | 0.74 |
| Root Mean Squared Error | 24480.31 | 25038.84 | 2.28 |
| R-squared | 0.83 | 0.824 | 0.72 |

### Decision Tree Regression


- On the right side, the decision tree regression chart, before any evaluation against training/test, shows the root node based on the total space built "totalSpcBlt". The final values at the leaves (on the right side) represent the predictions
- Parameters were chosen manually using the Fibonacci series. After experimenting with different options, selected the combination of parameters with the best outcomes. For some combinations of parameters, the outcomes were considerably bad.

| Decision Tree Regression | | | |
|---|---|---|---|
| **Metric** | **Train** | **Test** | **Var %** |
| Mean Absolute Error | 15730.3 | 21173.9 | 34.6 |
| Root Mean Squared Error | 22503.2 | 30687.5 | 36.36 |
| R-squared | 0.857 | 0.736 | 14.12 |




## Analysis And Critical Evaluation Of The Results

- Run the code over 10 times without seed, the Linear Regression Model had better performance and results are more consistent in comparison with Decision Tree Regression. Apart from that, even though both models have good performance there is a remarkable difference in the results of the metrics between the Linear Regression model and the Decision Tree Regression model. We can appreciate that difference by looking at the red-brown table in the "experimental results" section where the errors are over 20% for the test models and an R-squared with almost a 12 % difference. The percentages are the comparison between Decision Tree Regression results and Linear Regression results. For example, for Mean Absolute Error, the difference between the results for the train model between Linear Regression and Decision Tree Regression is given by the percentage figure of 12.37.
- Regarding the Decision Tree Regression metrics outcomes; the residuals of both sets, train and test, are randomly distributed around the zero line which indicates that the model has no bias, however overfitting can manifest in other ways and the metrics aforementioned are better indicators to spot overfitting. In this line, the results for the metrics of the decision tree model clearly show the presence of overfitting. The performance on the training set is significantly better compared to the test set. This seems to be a common issue with decision trees as create complex trees that fit perfectly the data as well as the noise. It is worth to mention the drop in the R-squared from train to test models, the train model probably has learned both, the patterns and the noise of the training data leading to a poor performance on unseen data.
- Decision Tree Regression, parameters: The best combination was found manually, using the Fibonacci series as a guide. The final combination was 99 for maximum number of splits whereas 8 was for the minimum leaf size. Too low values in the maximum number of splits lead to underfitting as the model can't capture enough information and results in poor performance on the other hand too high values make the model not only learn well about the patterns but also to capture the noise driving to overfitting. The minimum leaf size has the opposite behaviour, too high values lead to underfitting, why?, too high values means too many instances in each leaf, this makes the model to focus on more general trends instead of capturing specific details and although this is good to prevent overfitting the model can miss many information when run on other datasets. Contrarily, too low values for the minimum leaf size makes the model to capture too much information from the training data set uncovering too complex patterns as well as noise and outliers and that is not good if we want to run the model on other datasets since won't perform well
- To improve this model, strategies such as pruning the decision tree, using cross-validation for parameter tuning, or switching to a more robust model like Random Forest may be considered. Additionally, exploring feature engineering or selecting other predictors might help to enhance the model's performance.
- The Lineal regression model. The plotted linear regression graph has 95% confidence bounds dotted by red lines. This line shows where the true line is likely to be 95% of the time. There are some outliers that could affect the overall model performance.
- In general, Lineal Regression's residuals have also a reasonable random distribution however as the units in the predicted values increase the dispersion of the residuals increases too, this can mean that the model's error variance is changing with the level of the predictors, that is to say, heteroscedasticity. Would be informative checking them by applying the test of Breusch-Pagan.
- Metrics performance for the Linear Regression model: Mean Absolute Error results suggest that the model has learned a good generalization of the training data set. Root Mean Squared Error, considering that is more sensitive than MAE to outliers the fact that there is not a big difference between the training and test data is indicative that outliers don't represent a problem for this model. R-squared is high for both models (train and test), which is a sign that the dependent variable is well explained by the predictors. Overall, the three metrics outcomes suggest a strong stability for the Lineal Regression model.
- As a final conclusion for the comparison of both models we can state that the Linear Regression model is more stable and has a better performance than Decision Tree Model. Why does this happen? Well, the answer relies on how the data was processed before building the models. The procedures followed are more aligned with the assumptions and requirements of the Lineal Regression models. Let's analyse in detail: applied normalization and standardization to some variables, this is beneficial for linear regression models however Decision Tree Regression models are scale invariant. Applied Lasso Regression for feature selection, a linear approach that might not affect to Decision Tree Model. Applied VIF, a crucial step for the Lineal Regression model that doesn't affect in the same way to Decision Tree.
- Considering the size (low number of records) of the dataset the ideal technique to use for checking the model's performance would have been cross-validation, for simplicity used hold-on.
- Features like "GrLivArea" or "TotalSpcBuilt" have a high correlation with the target variable. This is consistent with the real world, the bigger the house the more expensive. This fact shows the importance of carefully selecting the predictors.
- There are highly influential features that weren't taken into consideration. Eg "OverallQual". The purpose of the study was to consider only tangible features. The results probably would be different.
- The model still has many variables and there is potential room for reducing the total amount of features.

## Lessons Learned, Future Work, References

### Lessons Learned

- I consider one of the most valuable lessons learned is to understand the creation process of a model from scratch (EDA, cleaning, feature engineering, hyperparameter tuning, etc) with this in mind my future works on the matter will be without any hesitation more streamlined apart from the fact that knowing the creation of a machine learning model will allow me to estimate ending date for projects.
- During this journey I also learned that there are many ways of optimizing a model, selecting the best option can drastically change the outcome.
- More features it doesn't mean better results for both models. I started with twice of the features I ended up applying to the model.
- Linear Regression and Decision Tree Regression models are excellent complementary models to boost final predictions.

### Future Work

- Improving the Decision Tree Regression by taking a tailored approach to this model. Some examples could be the application of grid search for hyperparameter tunning or pruning to reduce the risk of overfitting.
- Independently of the chosen model examine thoroughly which techniques to apply to do the best of the model applied. In other words, consider the model's assumptions.
- Making a deeper study of the domain knowledge would help to give the right steps in the process of building a model.
- Using a more robust validation like cross-validation.
- In categorical variables with high variability, the application of ANOVA could clarify how the different categorical variables affect to the response variable.

## REFERENCES

1. [1] P. Mehta et al., 'A high-bias, low-variance introduction to Machine Learning for physicists', Physics Reports, vol. 810, pp. 1–124, May 2019, doi: 10.1016/j.physrep.2019.03.001.
2. [2] D. De Cock, 'Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project', Journal of Statistics Education, vol. 19, no. 3, p. 8, Nov. 2011, doi: 10.1080/10691898.2011.11889627.
3. [3] J. I. Daoud, 'Multicollinearity and Regression Analysis', J. Phys.: Conf. Ser., vol. 949, p. 012009, Dec. 2017, doi: 10.1088/1742-6596/949/1/012009.
4. [4] P. Jadhav, V. Patil, and S. Gore, 'A Comparative Study of Linear Regression and Regression Tree', SSRN Journal, 2020, doi: 10.2139/ssrn.3645883.
5. [5] Z. Zhang, 'Decision Trees for Objective House Price Prediction', in 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), Taiyuan, China: IEEE, Dec. 2021, pp. 280–283. doi: 10.1109/MLBDBI54094.2021.00059.
6. [6] J. R. Quinlan, 'Induction of decision trees', Mach Learn, vol. 1, no. 1, pp. 81–106, Mar. 1986, doi: 10.1007/BF00116251.
7. [7] R. P. Ribeiro and N. Moniz, 'Imbalanced regression and extreme value prediction', Mach Learn, vol. 109, no. 9–10, pp. 1803–1835, Sep. 2020, doi: 10.1007/s10994-020-05900-9.
8. [8] H. Yu and J. Wu, "Real Estate Price Prediction with Regression and Classification CS 229 Autumn 2016 Project Final Report." Available: https://cs229.stanford.edu/proj2016/report/WuYu_HousingPrice_report.pdf
9. [9] S.-W. Choi, 'The Effect of Outliers on Regression Analysis: Regime Type and Foreign Direct Investment', QJPS, vol. 4, no. 2, pp. 153–165, Jun. 2009, doi: 10.1561/100.00008021.
10. [10] "MATLAB Documentation - MathWorks United Kingdom." https://uk.mathworks.com/help/matlab/index.html?s_tid=hc_panel