So let’s talk about our methodology. First of all we should explore and preprocess our data. That means we :

1. Visualized the data by plotting features against the house price to look for correlations and trends.

2. Generated descriptive statistics for each feature by calculating the mean, median, mode, range, standard deviation, and other measures.

3. We had to look for outliers in the data set.

4. And examine the data for any missing values or incorrect data.

Data Preprocessing:

1. Clean the data by removing any incorrect or missing values.

2. Standardize the data by scaling features to a common range.

3. Transform the data by applying transformations such as logarithmic, polynomial, and exponential functions in order to eliminate some of the outliers.

Then we should see how our features interact with each other

and combine two or more features to create a new feature.

and also select the most relevant features from a dataset to train the model.

Moreover, we trained our models in our train dataset and we used the test dataset to make our predictions.

For evaluation we used the coefficient of determination (R² or r-squared) which is a statistical measure that determines the proportion of variance in the dependent variable that can be explained by the independent variable.

As we can see the Gradient Boosting had the best results with a R² of 84%.

In essence, all of the known AI algorithms function like a black box. We provide them with certain information as input, and they provide an output. So a huge problem is that there is a tremendous trade-off between accuracy and interpretability. For instance, the linear regression model is simple to understand yet frequently has poor accuracy. Artificial Neural Networks, on the other hand, have a high degree of accuracy but are challenging to understand, making people less inclined to believe the results they provide. As the last part of our methodology we want to implement a new type of Artificial Intelligence called Explainable AI that will help us make our predictions more interpretable. Explainable AI comes to close that gap, that trade-off and that’s why it is so crucial to have it in our toolbox.

So in general Explainable AI (XAI) is a type of artificial intelligence that is designed to be more transparent and explainable to humans. XAI systems are designed to explain how they make decisions by providing explanations of their reasoning process, rather than merely providing a numerical result. By providing humans with a better understanding of how the system works, XAI can help increase trust and provide insights into potential bias in the system.

One of the most common algorithms of XAI is SHAP.

SHAP is a mathematical method to explain the predictions of machine learning models. It is based on the concepts of game theory and can be used to explain the predictions of any machine learning model by calculating the contribution of each feature to the prediction. For each feature, SHAP value explains the contribution to explain the difference between the average model prediction and the actual prediction of the instance. It also explains the contribution of different combinations of features.

We haven’t implemented this yet. So I will give you a different example in order to help you understand this.

Let’s think of a stroke data set for example we are interested in the contribution of the feature Age and when i say feature i mean a specific feature value. So for instance we have a value of 70 here. As input for this calculation we have the black box model f as well as an input data point x. This data point would be a single row in a tabular data set such as shown by this example. Now the first thing we do is iterate over all possible subsets set prime so combinations of features to make sure that we account for the interactions between our individual feature values. The reason why our sampling space is denoted with x prime here is because for more complex inputs like images, we of course don't treat each pixel as a feature but instead summarize them in some way. Using a mapping function we can then transform x to x prime but this is not really relevant in our example so one of those subsets could be for instance a subset of age and body mass index. This means we only consider to have information for those two, we don't know the values of, gender and heart disease and the other features. And now the most important step, we get the black box model output for this subset with and without the feature we are interested in. So in our example the difference in those two tells us how Age contributed to the prediction. For example the black box model output with Age would be seventy percent stroke and without age only ten percent. That means Age contributes sixty percent. That's also called the marginal value or SHAP value and then we do this for each possible combination.

So here are our SHAP outcomes. This vision assists us in two ways. First and foremost, we understand how much each feature contributed to the forecast. As you can see, the most crucial component for our forecasts was median\_income. Second, we now know that if we encounter the difficulty again, we should explore dropping some variables that appear to have little effect on the forecast or experimenting with new characteristics.

The relevance of the feature diminishes from top to bottom in the visualization. And how should we understand the model?

The price of a house rises as the median income rises. If the longitude lowers, the house's value rises, and so on.

This is advantageous because if someone want to purchase a house, he now knows not just the price projected by the model, but also the elements that contributed to that prediction and how much each item contributed to that projection.

To say the truth we faced some challenges. First of all we were amazed that our LSTM model didn’t perform well as they do usually. That may be because of a few different reasons. Firstly, if the data set was too small for the model to learn from, this could have caused the model to underperform. But we think that we have a very big dataset actually. But maybe isn’t enough. Who knows. Additionally, if the data set was not normalized or preprocessed properly, this could have caused the model to perform poorly. Finally, if the hyperparameters of the model were not tuned properly, this could have caused the model to not be able to learn properly and thus not perform well.

And the last reason is that LSTM may not be the best choice for this kind of problems.

Second, this isn't really a challenge. However, if we utilized a different dataset, we might need to reconsider our features. As an illustration. If we are in California, the pricing is heavily influenced by the ocean as a feature. However, if we are in Chicago, I feel the most important feature would be criminality.

Finally, the computing capacity of Google Colab proved insufficient for several algorithms. Especially for the random forest because we employed grid search, which can be computationally expensive. Grid search necessitates the algorithm fitting the model numerous times with different parameter combinations, which can be time expensive.