House value prediction

Executive summary:
Accurately assessing the value of a property holds significant importance for individuals, families, and real estate enterprises alike.

When engaging in transactions involving residential real estate, such as buying or selling a property, determining its value is imperative to establish a fair market price. Furthermore, the valuation of a property directly impacts the annual tax obligations of its owner.

House value prediction

Market analysis and objective:
Numerous platforms are available in the market, such as

- the Zillow estimator or,
- The estimator from different banks, e.g., Chase, or Bank of America,

enabling users to estimate the property values by considering the property's location.

The objective of the house value prediction project is to harness the power of predictive models to enhance the accuracy of forecasted values incorporating additional property descriptors such as the number of bathrooms, bedrooms, property size, and building year holds promise for refining tax predictions.

House value prediction

Value proposition and market segmentation:
The project aims to develop a platform to asses home value for

- Homeowners to achieve:
  Simplified house value estimation, key factors influencing the property value, and informed decision-making.
- Real Estate Professionals to support in:
  Enhanced service offerings to clients, increased customer satisfaction, and competitive advantage.

Through the introduction of this offer on our platform, our objectives are manifold:

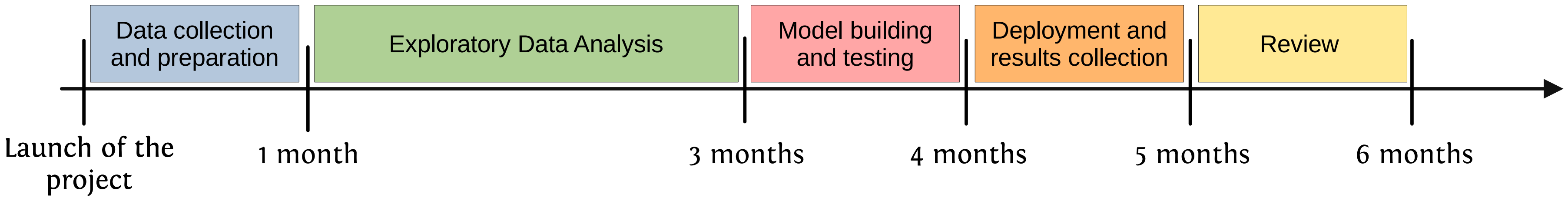
- To augment client contacts by 50% and enhance client acquisition by 10%,
- To refine the company's strategy towards targeting high-demand properties,
- To elevate the quality of service delivered by our real estate professionals,
- To assess the impact of online offers with the potential for launching new online platforms

House value prediction

Constraints:

- Time frame:
  The pilot project need to deployed within 6-months for the given budget
- Budget:
  It needs to account for the data acquisition and the engineering work

Milestones:



Business Impact Metrics:

- Enhancement of client outreach via the online application
- Alteration in the ratio of contacts converted to contracts
- Acquisition of data facilitated by the online application

Model evaluation

We investigated the following regression models:

- LinearRegression
- DecisionTreeRegressor
- KneighborsRegressor
- XGBRegressor
- GradientBoostingRegressor

To facilitate efficient model evaluation with our DataFrame containing nearly 3 million records, we opted to randomly downsample it. This ensured that our cross-validation routine ran within a reasonable timeframe.

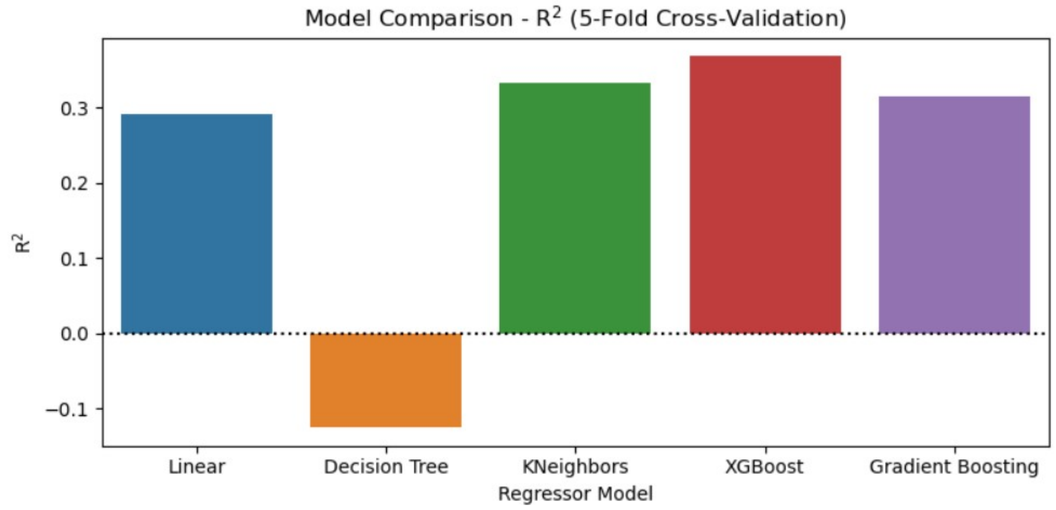
```
# Randomly sample 5% of the original dataframe
df_percent = df_filter.sample(frac=0.1)
print('The original df has shape:  %s' % (df_filter.shape,))
print('and the new df has shape: %s' % (df_percent.shape,))
```

```
The original df has shape:  (2950951, 11)
and the new df has shape: (295095, 11)
```

Model evaluation

The models were tested using the KFold cross validation method with 5 splits.
The evaluation of the models has been done using 4 metrics:

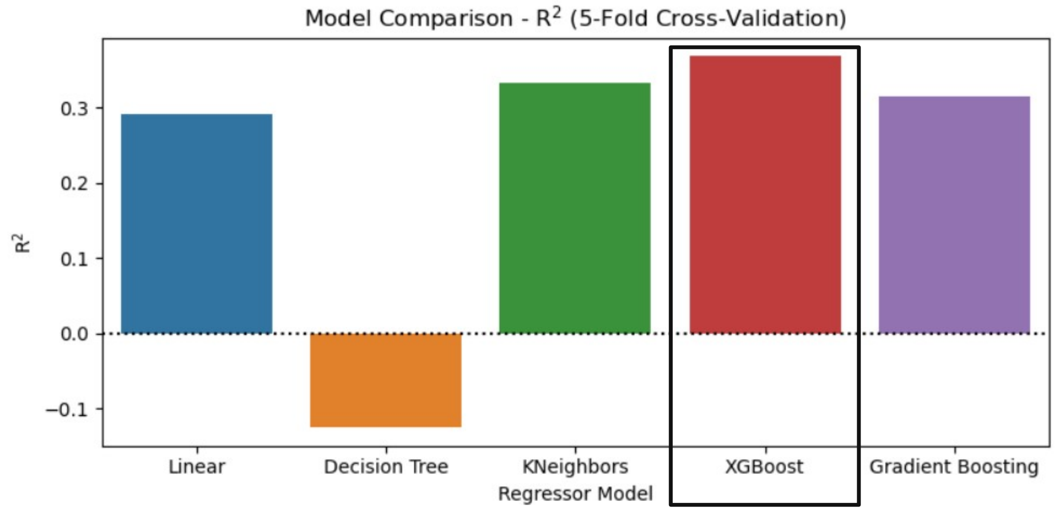
- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- R^2



	MAE	MSE	RMSE	R^2
LinearRegression	236653.059690	3.691842e+11	607605.309726	0.291343
DecisionTreeRegressor	251884.503199	6.357236e+11	797322.784559	-0.124568
KNeighborsRegressor	212247.988627	3.448101e+11	587205.348592	0.333305
XGBRegressor	209021.822484	3.274480e+11	572230.759519	0.367848
GradientBoostingRegressor	197537.805920	3.632510e+11	602703.115540	0.314460

Model evaluation

The best model, was then used to train the entire dataset



NN Model

We trained also a simple Neural Network with 4 dense layers

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 64)                704

 dense_1 (Dense)             (None, 32)                2080

 dense_2 (Dense)             (None, 16)                528

 dense_3 (Dense)             (None, 1)                 17

=================================================================
Total params: 3,329
Trainable params: 3,329
Non-trainable params: 0
_________________________________________________________________
```

However, the model did not outperform the XGBoost model, achieving and R^2 of 0.199