Sanjit Arunkumar (sanjita3), Arjun Rajesh Kenath Panikkath (ak85)

HOUSE PRICE PREDICTION

This is a project to perform house price prediction using machine learning techniques. In the following sections we explain step by step how we did the same.

Approach:

The data consists of 79 input features of a house that determine the output sales price value. A dataset this large requires a series of tasks to finally arrive at the prediction. Therefore, the following are our steps to perform the house prediction.

First we explore data analytics techniques including feature engineering, data cleaning to trim the dataset and keep only the relevant features and filling missing information. This includes taking a look at the data at a high level using data visualization techniques and gaining insight to decide how to modify the data and arrive at a dataset that is fit for training a machine learning model on.

Our next step is to choose a machine learning technique to perform the sales price prediction. This requires a regression model of some form such as linear regression, tree based regression models, neural network based regressors. In our approach, we have chosen a tree based model i.e XGBoost. The model has various parameters which can be modified to get least loss while training. Once the model parameters are fixed we fit the XGBoost Regressor on the training data.

We use the RMSE error on the predicted price and the actual price as a metric to measure the performance of the model.

Implementation Details:

List of packages used:

pandas
matplotlib
seaborn
xgboost
sklearn (RandomizedSearchCV)

Following are the steps in implementing the project:

1. Looking at the data to identify missing values. We do this using seasborn's heatmap function.
2. Fill the missing categorical values with the most common value.
3. Fill the missing continuous values with the mean value.
4. Drop the columns which have more than 50% of values missing.

5. Used XGBoostRegressor with Randomized Search Cross Validation(4 fold). Modifying the parameters and set parameters from the best fitting regression model with the following parameters. Used RMSE for training error.
6. Following is a snippet of the code to show the parameters used.

```
(base_score=0.25, booster='gbtree', colsample_bylevel=1,
    colsample_bytree=1, gamma=0, learning_rate=0.1, max_delta_step=0,
    max_depth=2, min_child_weight=1, missing=1, n_estimators=900,
    n_jobs=1, nthread=None, objective='reg:linear', random_state=0,
    reg_alpha=0, reg_lambda=1, scale_pos_weight=1, seed=None,
    silent=True, subsample=1)
```

Finally we created the submission file for Kaggle with the predicted values for sales prices and uploaded it to Kaggle to get the testing RMSE.

Experiments:

We did a variety of experiments to see how the feature engineering techniques would influence the training error of the xgboost model. Modifying the parameters of the XGBoost model using RandomizedSearchCV was also explored.

We experimented along the following lines:

1) Dropping a categorical column with missing values vs filling a column with mode
2) Modifying the parameters in the XGBoostRegressor.

The following tables indicate the results of our experiments:

Modifying each of the parameters independently we get the following values

Training Error	Parameter Modified
0.1890	Depth of Tree = 1
0.1123	Depth of Tree = 2
0.1456	Booster = GBTree
0.1678	Booster = GBLinear
0.1233	Dropping columns for less than 50% data
0.1489	Keeping columns for less than 50% data (filling with mode)

In our final submission we got a testing RMSE error of 0.1364 and a ranking of 1511/4579.

The following is the link to it https://www.kaggle.com/ak4785/competitions?tab=active

Discussion:

We have come to the conclusion that the house price prediction task is highly influenced by the features present and the quality of the data as shown by the experiments above. By modifying the way we handle the data in the input features the output values change significantly.

In the future, we plan on using other regression methods such as using an artificial neural network to predict the prices.

Citations:

https://machinelearningmastery.com/xgboost-for-regression/
https://www.analyticsvidhya.com/blog/2021/05/pandas-functions-13-most-important/
https://towardsdatascience.com/feature-engineering-in-python-part-i-the-most-powerful-way-of-dealing-with-data-8e2447e7c69e