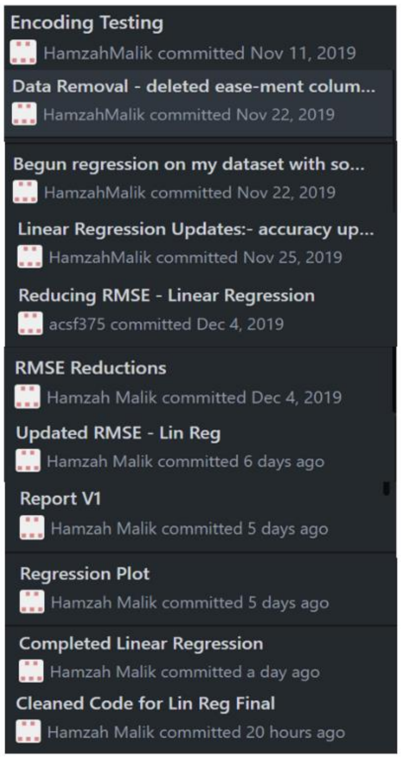
IN3062 – Introduction to Artificial Intelligence
Hamzah Malik ID: ACSF375

From our initial data cleaning, I helped filter out unwanted features. I suggested the removal of features such as 'ID' and 'Easement' as ID wasn't relevant to our results and Easement was an empty column, both of which were causing inflated RMSE values for our results. I also decided on what parameters we'd filter by, as there were hundreds of properties that had $10 as a sale price, as well as properties around the threshold of both $10 million and 2000 gross square feet- all of which contributed to skewed results. Filtering by these hyperparameters was fundamental in reducing the overall RMSE across all of our models.


Encoding Testing
HamzahMalik committed Nov 11, 2019
Data Removal - deleted ease-ment colum...
HamzahMalik committed Nov 22, 2019
Begun regression on my dataset with so...
HamzahMalik committed Nov 22, 2019
Linear Regression Updates:- accuracy up...
HamzahMalik committed Nov 25, 2019
Reducing RMSE - Linear Regression
acsf375 committed Dec 4, 2019
RMSE Reductions
Hamzah Malik committed Dec 4, 2019
Updated RMSE - Lin Reg
Hamzah Malik committed 6 days ago
Report V1
Hamzah Malik committed 5 days ago
Regression Plot
Hamzah Malik committed 5 days ago
Completed Linear Regression
Hamzah Malik committed a day ago
Cleaned Code for Lin Reg Final
Hamzah Malik committed 20 hours ago

I also helped with encoding our data by initially attempting to one-hot encoding. Whilst this technically worked, it also led to our headers becoming numeric too. Further research helped me find that categorical encoding was better suited for our data, something agreed upon by my team.

At the beginning of our project, I suggested it would be a good idea if we individually worked on separate models, splitting our models up amongst us. I decided on applying a linear regression model onto our data but before I started, I thought it would be a good idea to first visualise the correlation between our features.

I researched into using heatmaps within the seaborn data visualisation library, and plotted a correlation map using the following code: "correlation = df.corr() , sns.heatmap(correlation) , correlation['sale_price'].sort_values(ascending=False)" The heatmap proved to be crucial in the development of our models as it was what first highlighted to us that our dataset may be less linear than we had thought. I investigated further, using a pairplot chart within the seaborn library to visualise what features correlated with sale price, so we could discover if our most relevant features had a strong correlation to sale price (the results of which are in the report). Regular Bitbucket commits allowed my team-mates to review the visualisations above and any subsequent work I had completed.

Whilst implementing linear regression had its fair share of programming problems encountered, the real issue was trying to reduce my RMSE value. I had initially implemented linear regression without any filters, only to receive an obtuse RMSE value that was 900% of our mean. The data filters mentioned at the top of my reflection once implemented, had managed to reduce this value down significantly, but it was still quite high. Filtering the features we used to plot against our predicted sale price also helped reducing our RMSE, by using: "df = df.select_dtypes(include=['int32', 'int64', 'float'])". This ultimately led to the RMSE of linear regression being 50% of our mean in total.

The reduction of RMSE of course reflected well on my diagrams, which had their 'Predicted' vs 'Actual' results much closer than at the beginning of my attempts. I had also plotted a 'Regression Plot' as shown in the 'Linear-Regression.py' file, using unknown data to predict our Y values.

As expected, linear regression yielded the highest RMSE value among our models. Despite this, I felt as though I made sensible choices throughout the project to help reduce this value, including: experimentation with our hyperparameters, filtering out noisy irrelevant data and unrelated features. Our RMSE had started off at over 900% and had managed to settle it at roughly 50%. Whilst I would have liked to reach around 25% RMSE, I had already pushed the capabilities of my model to its limits and don't think I would have achieved better results in return for what it would have cost (even further reduction to the size of our dataset.)