Foundations of Data Science (CS F320)

Assignment – 2

BITS Pilani- Hyderabad Campus

Members:

- Akshat Agrawal - 2019AAPS0264H
- Kshitij Upadhyay - 2019A7PS0105H
- Umesh Kumar A- 2019A4PS0868H

Brief Description Of The Model

Model Used:

The model used for our project is Linear Regression, and the optimization algorithm used is Gradient descent.

Linear Regression:

In a Linear Regression, We have the following equation with a single predictor:

$$Y = \theta_0 + \theta_1 x$$

where,

Y is the target,

x is the predictor,

θ_0 *is the bias,*

and θ_1 is the weight in the regression equation

This linear equation can be used to represent a linear relationship.

Data Insights

Feature Number	Feature	Mean
1	bedrooms	3.382997
2	bathrooms	2.143098
3	sqft_living	2124.656729
4	sqft_lot	15797.568182
5	floors	1.509787
6	waterfront	0.003367
7	view	0.245791
8	condition	3.425926
9	grade	7.687710
10	sqft_above	1815.946337
11	sqft_basement	304.373737
12	sqft_living15	2019.319865
13	sqft_lot15	12889.924242

Expectations

Looking at the data, we are quite sure feature selection will play a role in building an optimal model. There are feature which look quite similar and hence the covariance between them will be quite high so we will have to drop them. As specified, we will use forward and backward greedy feature selection for electing the optimal pool of features.

Moreover the data has quite a few missing values which we will replace with mean of that feature.

Methodology

Greedy selection is a popular technique for feature subset selection. The main advantage of this approach is its simplicity and generally low run-time in small feature spaces. This makes greedy selection applicable to many practical problems.

We loop over all the features and select the one with the best model. With the feature selected, we loop over all features but the one selected and compare the testing results. If any model performs better than the previous one, we select that feature into our pool, otherwise stop our selection procedure.

On the contrary in the backward greedy selection procedure, we take all the features in the pool and try to eliminate them one by one. This time we remove one feature at a time, rather than adding it one at a time.

Part A: Greedy Forward feature selection

Below we have tabulated the testing error achieved by our model and feature selected by using linear regression to predict the output.

Iteration	Feature Selected	Root Mean Square Testing Error	Root Mean Square Training Error
1	3	381370.7371767867	223686.8512829784
2	9	373537.8830258457	215883.48053015347
3	6	367521.09843689227	213599.72741573423
4	11	364873.8049596437	212794.86519071614
5	10	360027.0667702455	212302.66872632183
6	1	357569.3056895571	212046.91950266017
7	8	356836.0196123118	209944.65752204714
8	13	356081.4398225572	209392.64082501535

Part B: Greedy Backward feature selection

Below we have tabulated the testing error achieved by our model and feature removed by using linear regression to predict the output.

Iteration	Feature Removed	Root Mean Square Testing Error	Root Mean Square Training Error
1	7	356955.8264436319	207086.97873766618
2	12	354322.8928717752	207942.40751357333
3	4	353594.9852346561	209421.26399326374
4	5	353054.6884300014	209642.16273306622
5	2	352854.96431871684	209913.6169308863

Part C: Linear regression model without any pre-processing and feature selection

Root Mean Square Testing Error	Root Mean Square Training Error
363500.4747200253	199812.44971546254

Conclusions

We observe that in the forward greedy algorithm, we get feature numbers **1,3,6,8,9,10,11 and 13**, namely **bedrooms, sqft_living, waterfront, condition, grade, sqft_above sqft_basement, sqft_lot15.**

On the other hand we get feature number **1,3,6,8,9,10,11 and 13** in the backward greedy algorithm for feature selection namely **bedrooms, sqft_living, waterfront, condition, grade, sqft_above, sqft_basement, sqft_lot15.**

Both the results match, hence strengthening the claims of feature selection we have made.

The feature selection works and is required because of the features that seem to be quite similar. For example feature 3 and 12, 4 and 13 seem to be quite similar, both in terms of what they represent and the values they posses, ie. they share **high covariance**.

We evaluated our model on evaluation metrics **Root Mean Square Error**, which appears to have quite done the job.

From the third part where we ignored any form of normalisation and feature selection we yet again confirm the importance of normalization and feature selection. The Errors appear to be **quite high in both the training and testing.**