CS F320 : FODS

Assignment 2

Date : 7th December, 2021

Team Members

R Adarsh	2019A7PS0230H
Abhinav Talesra	2019AAPS0223H
Sahaj Gupta	2019A7PS0148H

Problem 1 Linear Regression with Feature Subset Selection

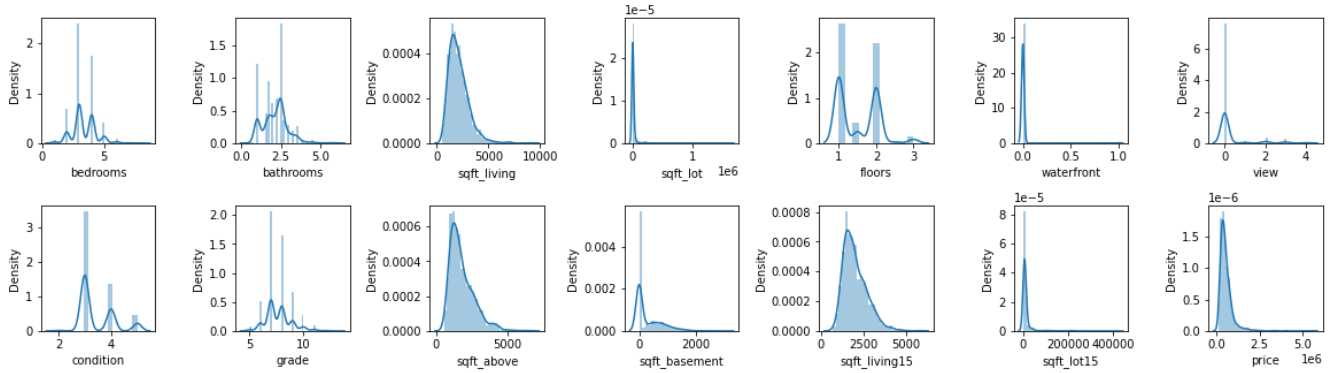
1.1. Model Description and implementation

Dataset given:

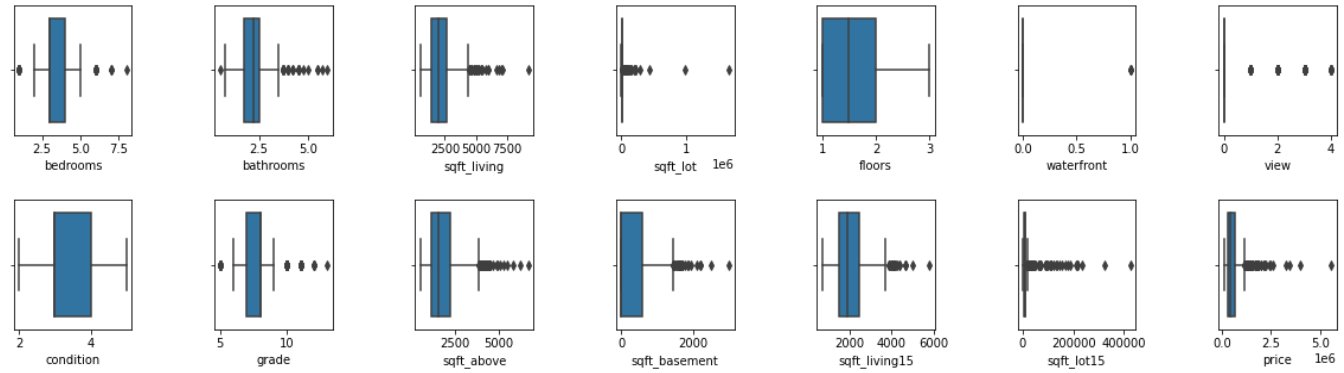
The given dataset consists of 1188 data points, with 13 features and rent as the target attribute. There are some missing values under the features : sqft_living, floors,sqft_above.

Visualizing outliers in data:

A) Feature Vs Density plots



B) Feature Box plots



From the above density plots and box plots we can see that there are some outliers under every feature.

Data Processing:
Data processing is the collection and manipulation of items of data to produce meaningful information. Pre-process your data includes :

- Shuffling the data
- Standardizing/normalizing the values

$$x_{\text{stand}} = \frac{x - \text{mean}(x)}{\text{standard deviation }(x)} \qquad x_{\text{norm}} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

- Creating a random 70-30 split to aid in training and testing respectively.
- Handling missing values: The data points having missing feature values can either be dropped entirely before training the model or the missing values can be imputed with mean/median/mode of feature values depending on the nature of the feature.
- Detecting and visualizing outliers

Loss Function:
Loss function or cost function is a function that maps an event or values of one or more variables onto a real number intuitively representing some "cost" associated with the event. The error function used for regression task is mean squared error given by the formula,

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Optimization:

Our objective is to find the optimum set of parameters that minimize the cost function. We have Gradient Descent and Stochastic Batch gradient descent as optimization algorithms.

A) Standard/ Batch Gradient Descent:

1. We start by initializing random weights
2. In each iteration we update the parameters,

$$\theta_j \leftarrow \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta)$$

$$\frac{\partial}{\partial\theta_j}J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)}$$

$$\theta_j \leftarrow \theta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)}$$

3. By iterating over the training examples until convergence we are able to reach the optimum parameters that minimize the cost.

Model Description:

Linear Regression is a linear approach to model the relationship between the features and the target attribute. The case of linear regression dealing with more than one exploratory feature is called multiple linear regression.

Greedy Forward Feature Subset Selection:

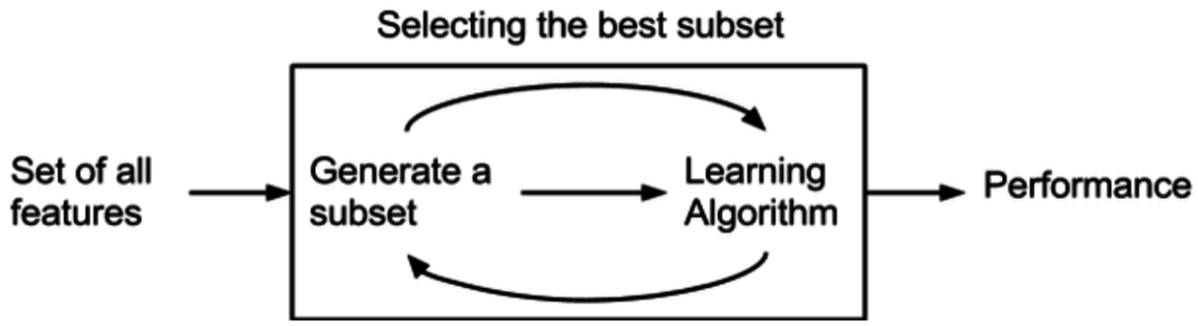
This method iteratively finds the best new feature to be added to the subset .

1) We initially start with zero features and then find a feature that minimizes the RMSE score when the model is trained/tested on this feature.
2) Repeat by selecting and adding a new feature greedily in each iteration.
3) Stop when our feature_subset = set of original features

Greedy Backward Feature Subset Selection:

This method greedily removes features from the set of features iteratively .

1) We initially start with n features and then find a feature, which when eliminated from the set would minimize the RMSE score when the model is trained/tested.
2) Repeat by removing a feature greedily in each iteration.
3) Stop when our feature_subset becomes empty



1.2 The final train and test metrics (lr = 0.0005 || epochs = 10000)

With standardization:

Method	RMSE_train	RMSE_test
Forward Selection	0.637	0.639
Backward Selection	0.620	0.625
No feature selection	0.619	0.666

With normalization:

Method	RMSE_train	RMSE_test
Forward Selection	0.046	0.0417
Backward Selection	0.044	0.0419
No feature selection	0.043	0.044

1.3. Results

With Standardization:
The subset of features that gives the best performance selected from the data using:

a) Greedy forward feature selection:
`['sqft_living', 'grade', 'waterfront', 'condition', 'sqft_lot15', 'bathrooms', 'floors', 'bedrooms']` gives rmse_train=0.637 and rmse_test=0.639.
b) Greedy backward selection:
`['bedrooms', 'waterfront', 'view', 'condition', 'grade', 'sqft_above', 'sqft_basement', 'sqft_lot15']` gives rmse_train=0.620 and rmse_test=0.625 .
c) Without feature selection: Model trained with all features gives rmse_train = 0.619 and rmse_test = 0.666 .

With Normalization:
The subset of features that gives the best performance selected from the data using:

a) Greedy forward feature selection:
`['sqft_living', 'sqft_above', 'bedrooms', 'view', 'sqft_lot15', 'floors', 'condition', 'bathrooms', 'sqft_lot']` gives rmse_train=0.046 and rmse_test=0.0417.
b) Greedy backward selection:
`['bedrooms', 'floors', 'view', 'condition', 'grade', 'sqft_above', 'sqft_basement', 'sqft_living15', 'sqft_lot15']` gives rmse_train=0.044 and rmse_test=0.0419 .
c) Without feature selection: Model trained with all features gives rmse_train = 0.043 and rmse_test = 0.044

1.4 Conclusion

Generally, in real-world datasets we may have to deal with huge amounts of data with high dimensions. Many times, this may lead to performance problems while training and may affect the accuracy of the model. Thus, feature subset selection aims to select a subset of features which gives an optimal model and reduces the dimension of the data points. It is an important preprocessing step in data engineering.

From the results obtained , it is evident that we get lower rmse scores on test sets with greedy feature subset selection techniques as compared to models without any feature selection. So it can be concluded that not all features contribute to the performance of the model and a subset of 8-9 features contain the essential information to predict the rent prices of houses. These features have marginal importance over the others in predicting the rent price of the house and provide the optimal regression model for the given data set.