VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Department of Computer Engineering

Report on

House Price Prediction System

Submitted by:

Paresh Kalinani (25)

Abhishek Mehta (33)

Dinesh Purswani (47)

Roshan Rajwani (48)

Class: D17C

Table of Contents

PROBLEM STATEMENT:

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad. But this playground competition's dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence.With 28 explanatory variables describing (almost) every aspect of residential homes in this competition challenges you to predict the final price of each home.

1. Introduction

Machine learning is a field of computer science that gives computer systems the ability to "learn" with data, without being explicitly programmed.
Machine learning has been used for years to offer image recognition, spam detection, natural speech comprehension, product recommendations, and medical diagnoses. Today, machine learning algorithms can help us enhance cybersecurity, ensure public safety, and improve medical outcomes. Machine learning systems can also make customer service better and automobiles safer.

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad. But this playground competition's dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence.With 28 explanatory variables describing (almost) every aspect of residential homes in, this competition challenges you to predict the final price of each home.

We work on a dataset which consists of information about the location of the house, price and other aspects such as square feet etc. When we work on these sorts of data, we need to see which column is important for us and which is not. Our main aim is to make a model which can give us a good prediction on the price of the house based on other variables. We are going to use Linear Regression for this dataset and see if it gives us a good accuracy or not.

In this case, we can have both clients with no conflict of interest!

Client House buyer: This client wants to find their next dream home with a reasonable price tag. They have their locations of interest ready. Now, they want to know if the house price matches the house value. With this study, they can understand which features (ex. Number of bathrooms, location, etc.) influence the final price of the house. If all matches, they can ensure that they are getting a fair price.

Client House seller: Think of the average house-flipper. This client wants to take advantage of the features that influence a house price the most. They typically want to buy a house at a low price and invest on the features that will give the highest return. For example, buying a house at a good location but small square footage. The client will invest on making rooms at a small cost to get a large return.

2. Dataset Used

The dataset used for the following project is House Price Prediction dataset, readily available on Kaggle.com:
It has the following data fields:

- MSSubClass: The building class
- LotArea: Lot size in square feet
- PoolArea: Pool area in square feet
- OverallQual: Overall material and finish quality
- OverallCond: Overall condition rating
- YearBuilt: Original construction date
- YearRemodAdd: Remodel date
- MiscVal: $Value of miscellaneous feature
- BsmtFinSF1: Type 1 finished square feet
- BsmtFinSF2: Type 2 finished square feet
- BsmtUnfSF: Unfinished square feet of basement area
- TotalBsmtSF: Total square feet of basement area
- 1stFlrSF: First Floor square feet
- 2ndFlrSF: Second floor square feet
- LowQualFinSF: Low quality finished square feet (all floors)
- GrLivArea: Above grade (ground) living area square feet

3. Algorithm Used

Regression is a machine learning tool that helps you make predictions by learning – from the existing statistical data – the relationships between your target parameter and a set of other parameters. According to this definition, a house's price depends on parameters such as the number of bedrooms, living area, location, etc. If we apply artificial learning to these parameters we can calculate house valuations in a given geographical area.

We have used the following algorithms:

Ridge Regression: Ridge Regression is an extension for Linear Regression. It's basically a regularized Regression model. The λ parameter is a scalar parameter which should be learned using Cross validation.A super important fact about ridge regression is that enforces the β coefficients to be lower, but it does not enforce them to be zero. That is, it will not get rid of irrelevant features but rather minimize their impact on the trained model.

Lasso Regression: Lasso is another extension built on regularized linear regression, but with a small twist. The loss function of Lasso is in the form:

$L = \sum(\hat{Y}i- Yi)2 + \lambda\sum |\beta|$

The only difference from Ridge regression is that the regularization term is in absolute value. But this difference has a huge impact on the trade-off we've discussed before. Lasso method overcomes the disadvantage of Ridge regression by not only punishing high values of the coefficients β but actually setting them to zero if they are not relevant. Therefore, you might end up with fewer features included in the model than you started with, which is a huge advantage.

Ransac Algorithm: The RANdom SAmple Consensus (RANSAC) algorithm proposed by Fischler and Bolles is a general parameter estimation approach designed to cope with a large proportion of outliers in the input data. Unlike many of the common robust estimation techniques such as M-estimators and least-median squares that have been adopted by the computer vision community from the statistics literature, RANSAC was developed from within the computer vision community.

RANSAC is a resampling technique that generates candidate solutions by using the minimum number observations (data points) required to estimate the underlying model parameters. As pointed out by Fischler and Bolles, unlike conventional sampling techniques that use as much of the data as possible to obtain an initial solution and then proceed to prune outliers, RANSAC uses the smallest set possible and proceeds to enlarge this set with consistent data points.

Finally we decided to go ahead with the RANSAC algorithm as it was efficient and had less error.

4. Analysis

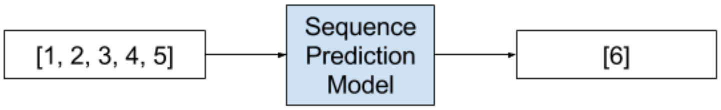
Our project is a type of prediction based Project.It uses prediction in machine learning to get approximate house price prediction.

Sequence prediction is different from other types of supervised learning problems.The sequence imposes an order on the observations that must be preserved when training models and making predictions.Generally, prediction problems that involve sequence data are referred to as sequence prediction problems, although there are a suite of problems that differ based on the input and output sequences.

Sequence prediction involves predicting the next value for a given input sequence.

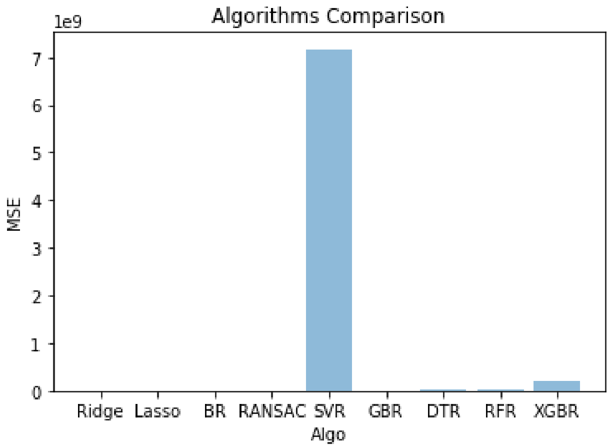
For example:

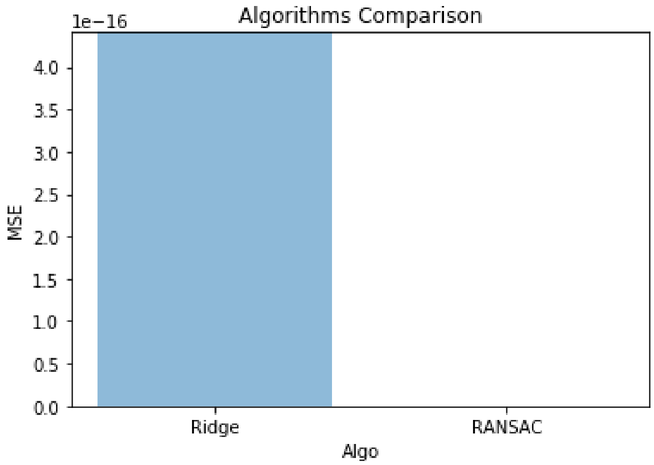
- Given: 1, 2, 3, 4, 5
- Predict: 6

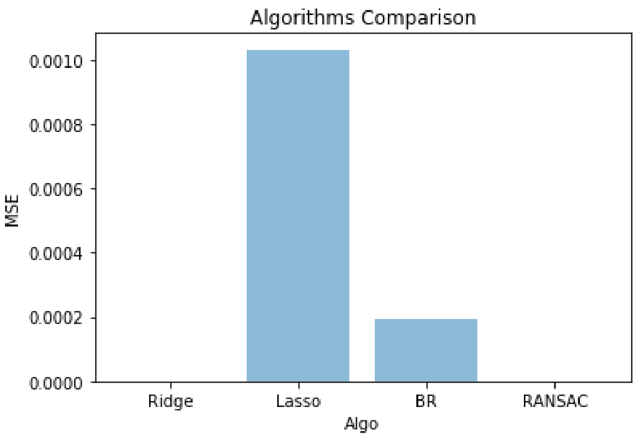


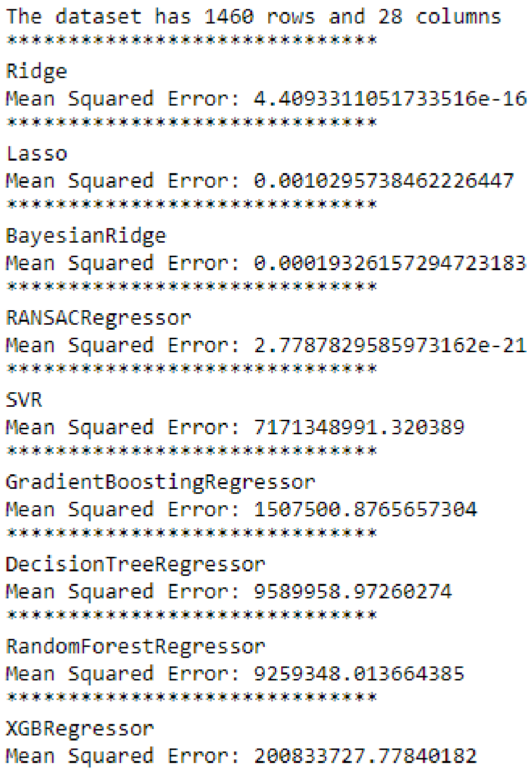
A prediction model is trained with a set of training sequences. Once trained, the model is used to perform sequence predictions.
A prediction consists in predicting the next items of a sequence. This task has numerous applications such as web page prefetching, consumer product recommendation, weather forecasting and stock market prediction.

5. Output

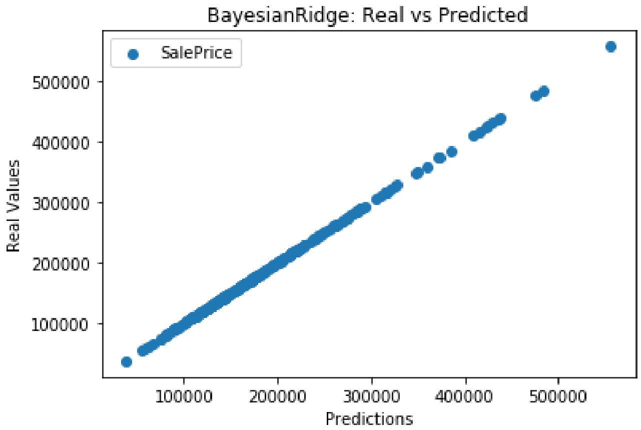


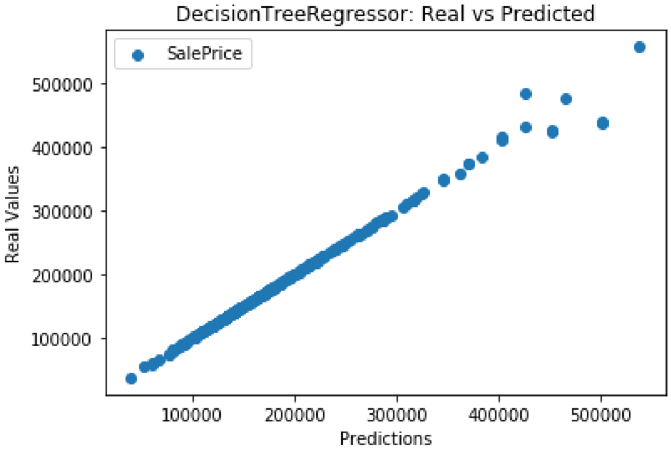


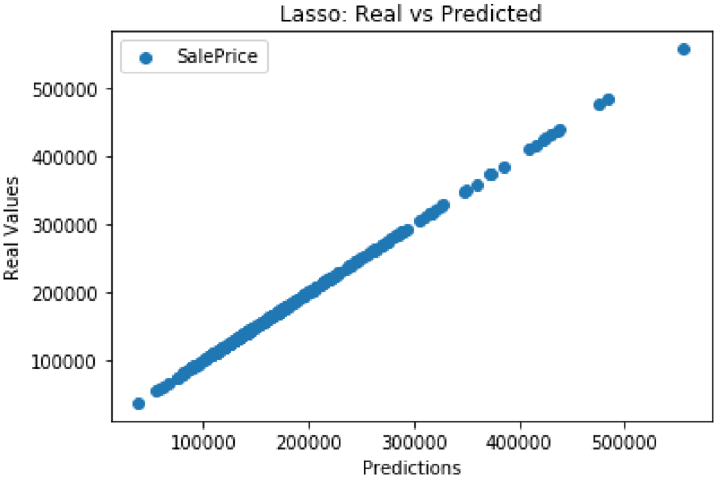


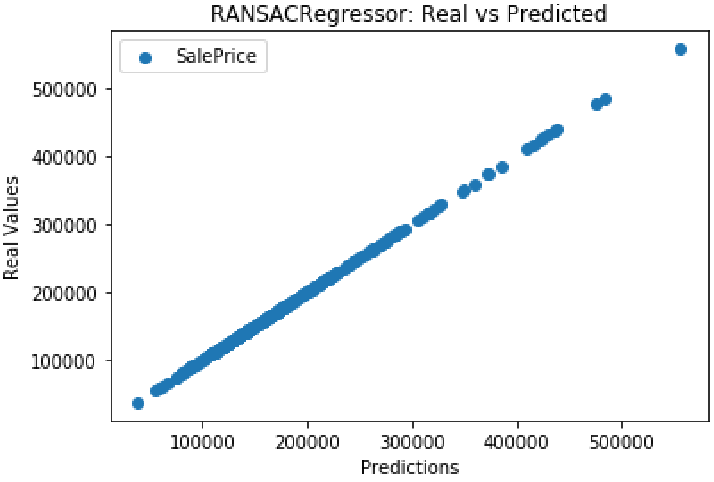


```
The dataset has 1460 rows and 28 columns
******************************
Ridge
Mean Squared Error: 4.4093311051733516e-16
******************************
Lasso
Mean Squared Error: 0.0010295738462226447
******************************
BayesianRidge
Mean Squared Error: 0.00019326157294723183
******************************
RANSACRegressor
Mean Squared Error: 2.7787829585973162e-21
******************************
SVR
Mean Squared Error: 7171348991.320389
******************************
GradientBoostingRegressor
Mean Squared Error: 1507500.8765657304
******************************
DecisionTreeRegressor
Mean Squared Error: 9589958.97260274
******************************
RandomForestRegressor
Mean Squared Error: 9259348.013664385
******************************
XGBRegressor
Mean Squared Error: 200833727.77840182
```




DecisionTreeRegressor: Real vs Predicted
SalePrice
Real Values
Predictions
100000
200000
300000
400000
500000
100000
200000
300000
400000
500000


Lasso: Real vs Predicted
SalePrice
Real Values
Predictions
100000
200000
300000
400000
500000
100000
200000
300000
400000
500000


RANSACRegressor: Real vs Predicted
SalePrice
Real Values
Predictions
100000
200000
300000
400000
500000
100000
200000
300000
400000
500000