Machine Learning Engineer Nanodegree Capstone Proposal

House Prices Prediction

Hesham Shabana December 18th, 2016

Proposal

Domain Background

House prices is a topic that always attract a lot of attention due to the daily demand and the wide range of interested party's investors, real estate agents, tax estimators, house owners, and house buyers with this the demand for a reliable model to assist a house price is needed. Traditionally house price prediction does not take into consideration the wide range of available parameters and it also assume independence between these parameters which does not hold in practice.

In Egypt buying a house is very important decision especially for young people who are looking to start a family and this decision becomes even harder with this increase in population, Egypt population estimated to be 93,383,574 with almost 2% growth in the last 4 years, and there is very little research in this area as well as a lack of data. Not to mention that people are following only one rule what is the square feet price in specific area? Therefore, using a machine learning algorithm to learn from the past purchasing history will help us to determine and predict the price of the house taking into consideration the attributes that matter the most and this will support any new buyer or a seller to set the right expectation.

Reference

- http://www.doc.ic.ac.uk/~mpd37/theses/2015_beng_aaron-ng.pdf
- https://docs.google.com/viewer?url=patentimages.storage.googleapis.com/pdfs/US6609109.pdf
- https://researcharchive.lincoln.ac.nz/bitstream/handle/10182/5198/House_%20price_%20prediction.pdf
- http://link.springer.com/article/10.1007/s11146-007-9036-8

Problem Statement

The goal is to analyze historical sales of house prices features and through feature selection, feature engineering, machine learning finds the most relevant set of features that affect the price and which will allow us to perform an accurate prediction for new houses.

To avoid curse of dimensionality given the large number of features included in our database (80 feature) first we need to reduce our vector space by applying feature selection methods, as well as feature transformation by leveraging algorithms like PCA, ICA, Lasso regression. The goal here is to reduce our input space as much as possible to avoid overfitting and to achieve better prediction results.

Datasets and Inputs

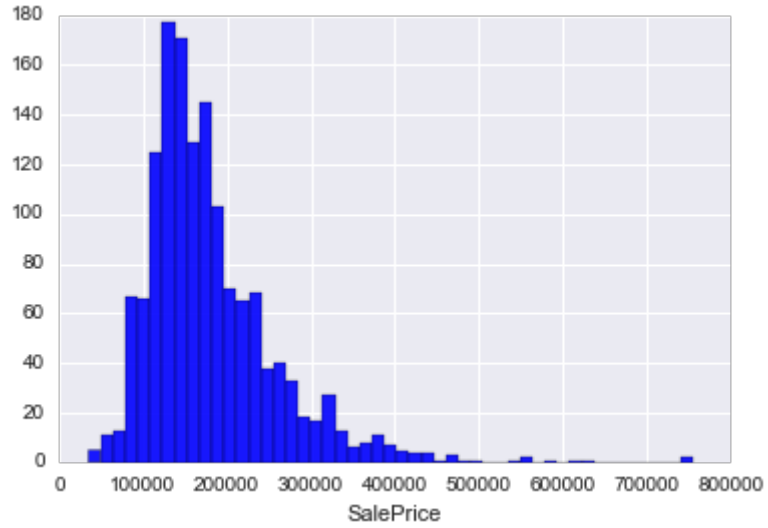
This dataset describing the sales for houses in Ames, Iowa from 2006 to 2010 which contains 2930 observations and 80 variables, the variable distribution is 23 nominal, 23 ordinal, 14 discrete and 20 continuous.

This dataset proves that much more influences price negotiations than the number of bedrooms or a white- picket fence.

Dataset Copyrights

Statistics on the Housing Price variable:

Housing price distribution

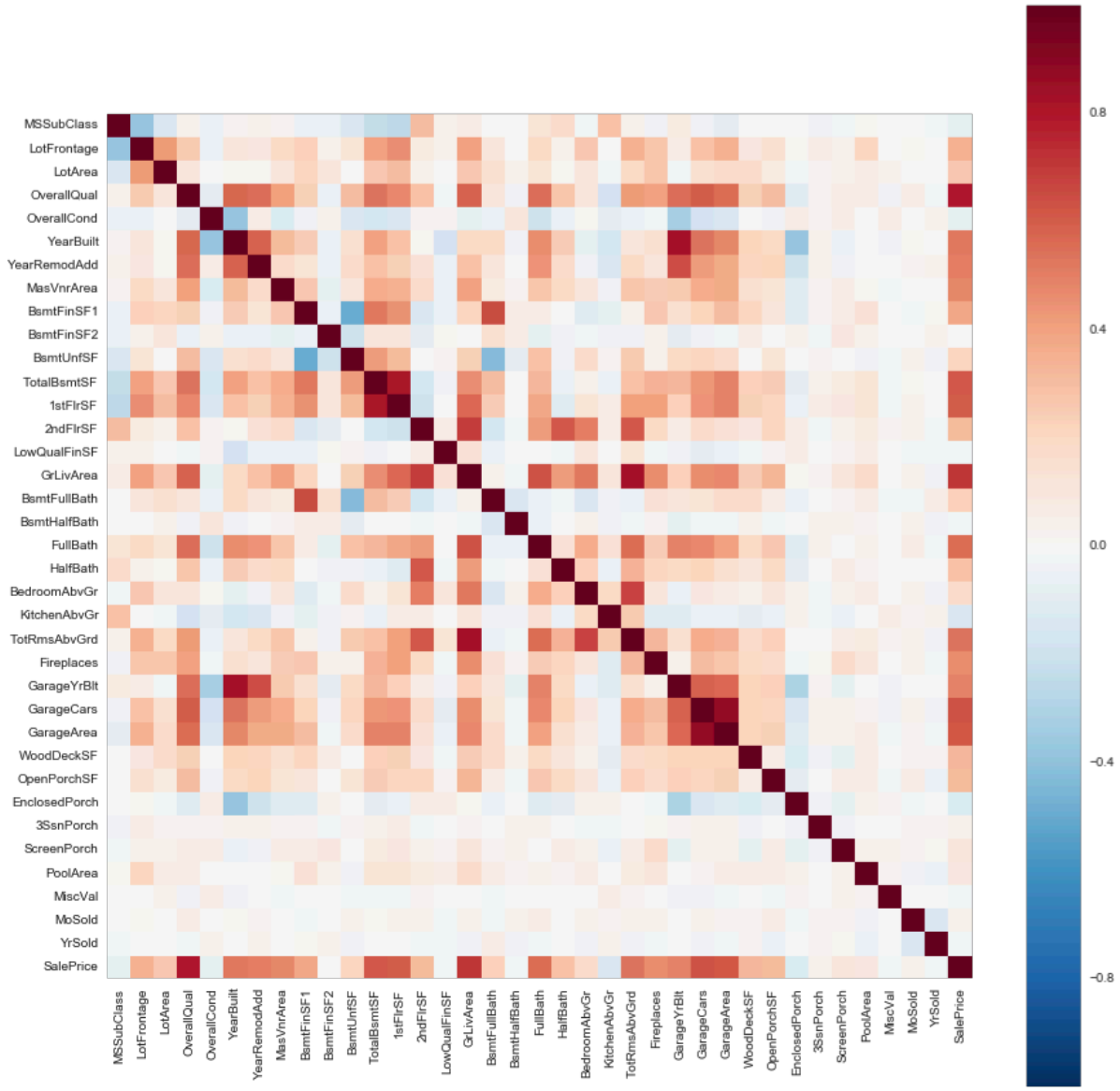


As we can see that our distribution for the target variable is positive skew "skewed to the right", therefore, transform our data to be normally distribute may yield a better result, Box Cox (http://www.itl.nist.gov/div898/handbook/eda/section3/boxcoxno.htm) transformations might be used.

Data features

Numerical features

The large number of continuous features (20) and discrete features (14) in this dataset give us various methods to combining and selecting the most relevant features.



Categorical features

This dataset has large number of categorical variables (23 nominal, 23 ordinal). They range from 2 to 28 classes with the smallest being STREET (gravel or paved) and the largest being NEIGHBORHOOD (areas within the Ames city limits). In addition, the nominal variables identify various types of dwellings, garages, materials, and environmental conditions while the ordinal variables typically rate various items within the property

```
['MSZoning' 'Street' 'Alley' 'LotShape' 'LandContour'
'Utilities''LotConfig' 'LandSlope' 'Neighborhood'
'Condition1' 'Condition2' 'BldgType' 'HouseStyle'
'RoofStyle' 'RoofMatl' 'Exterior1st' 'Exterior 2nd'
'MasVnrType' 'ExterQual' 'ExterCond' 'Foundation'
'BsmtQual' 'BsmtCond''BsmtExposure' 'BsmtFinType1'
'BsmtFinType2' 'Heating' 'HeatingQC''CentralAir'
'Electrical' 'KitchenQual' 'Functional' 'FireplaceQu'
'GarageType' 'GarageFinish' 'GarageQual' 'GarageCond'
'PavedDrive' 'PoolQC' 'Fence' 'MiscFeature'
'SaleType' 'SaleCondition']
```

Solution Statement

A solution for this problem is to take the input features and make them suitable to be used by a machine learning algorithms (*more details are provided in Project Design section*), in addition, to do feature selection and engineering to enhance the performance of the model, at the end we should have a list of the most relevant features and an accurate model that could be used to predict new house prices.

Benchmark Model

"*Traditionally house price prediction does not take into consideration the wide range of available parameters.*" Therefore, our benchmark model could be a linear regression model that predict the price based on some of the traditional parameters like square_feed, num_of_rooms, neighbor and assume independence between variables.

Evaluation Metrics

To assists the model performance, the below matrices will be used:

- **Bias:**

Positive values indicate the model tends to overestimate price (on average) while negative values indicate the model tends to underestimate price.

- **Maximum Deviation:**

Identifies the worst prediction made in the validation data set.

- **R^2 score:**

Feature Elimination: Measure the proportion of the variance that is predictable from another variable, which may help to remove a feature that has high correlation or explained by another featureModel evaluation: Measure of how well future samples are likely to be predicted by the model.

Best possible score is 1.0 Can be negative if the model doing worse

- **Mean squared error:**

Measures the average of the squares error, it is the difference between the estimator and what is estimated variable

Project Design

To solve this problem, the following steps shall be executed:

Data Preprocessing

- Feature Scaling VS. Feature StandardizationFeature Scaling for continues variables so they can be compared on compound

ground, scale down all the features between 0 and 1.
- Feature Standardization: rescale features so they have the properties of normal distribution, as PCA may perform better after standardization. Imputation of missing values
- One-Hot Encoding for categorical features Split the data to training and validation sets

Data Preprocessing Reference: (https://www.analyticsvidhya.com/blog/2016/07/practical-guide-data-preprocessing-python-scikit-learn/)

Features

Feature selection methods (Hill climbing, SelectPercentile, SelectKBest) Feature transformation (PCA algorithm, Lasso regression)Feature engineering

Training

Train our model using the below algorithms: Logistic regressionSVMBoosting

Random forest

Model tuning

Tune our model parameters by using GridSearchCV or RandomizedSearchCV

Evaluation

Evaluate the model performance using R^2 and MSE

Result

Apply our model to the testing data