MLND Capstone Project: Predicting Ames Housing Prices

Gendith M. Sardane

admin@adalace.org || Github:gmsardane

October 2, 2016

Abstract

As an extension and an updated version of the Boston Housing regression problem, we present our the results of our regression analysis using the Ames Housing dataset compiled...

1 Introduction

As a first project for the MLND, we were tasked to predict the prices of homes located in various suburbs in Boston, MA. The Boston housing dataset (Harrison and Rubinfeld 1978) was collected in 1978, and contains roughly 500 examples described by 14 features. To make them more realistic with today's home values, each entry was multiplicatively scaled to account for the $>$ 30-year gap in the housing data. Because of this 30-year gap, I was quite dissatisfied with the analysis and results. In my opinion, scaling by a multiplicative factor does not capture the intricacies of how a home is valued. As I am planning to buy a home very near in the future, I was naturally quite curious of how various aspects of a house factor into its final price tag. Needless to say, I am in search for a real data set rather than a "realistic" one.

For my capstone project, I wanted an updated version of such an example of a regression problem. As it turns out, Kaggle currently hosts a competition on predicting the prices that a house sells using the Ames housing dataset. The competition is live from August 30, 2016 through March 1, 2017.

In this analysis, I use the data provided in train.csv to train a gradient boosted decision tree to predict home values. In this report, I will present the results and the approach that I took in finding the best model that would best predict the prices of homes sold in Ames, IA from 2006-2010. In particular, this submission is organized as follows: In §2, I will give a brief background on the dataset. In §3, I discuss the Gradient Boosting regression. The statistical properties of some major features in the dataset, and the feature selection process will be presented in §4. I then present the main results in §5. In §6, I will present my conclusions and present a comparisons of the results of this analysis to that of the Boston housing project.

2 The Ames Housing Dataset

The main reference for the Ames housing dataset is due to De Cock (2011). In brief, the dataset is a compilation of attributes of more than 2900 properties sold between 2006 and 2010 in various locations in Ames, IA. The data originally came as a data dump from the City's Assessor's Office. As presented by De Cock (2011), the dataset comes as a collection of 80 features directly related to property sales. In particular, data consist of 23 nominal, 23 ordinal, 14 discrete, and 20 continuous explanatory variables. As with the Boston dataset, the target variable is the "Sale Price", which a continuous quantity representing the monetary value of the home in US dollars(USD).

For the purposes of the competition, Kaggle divides the dataset into two sets of roughly equal sizes. The first set (**train.csv**), which contains 1460 entries and 80 features, is intended for training. The second set (**test.csv**), which is meant for training, contains 1459 entries and 79 features, with the property sale price withheld.

Appendix A summarizes these quantities, grouped according to their respective data class, as well as the codes used for column nomenclature in the file train.csv, and whether these features were used in the final regression model.

3 The Gradient Boosting Regression

Gradient boosting (GB) regression is one type of ensemble machine learning technique for creating regression models consisting of "collections" of (weak) regressors, typically decision trees. Here, the learners are learned sequentially, as new models are fit for a more accurate estimate of the response variable. The new-base learners are "constructed to be maximally correlated with the negative gradient of the loss function" for the whole ensemble (Natekin & Knoll 2013). In general, the loss function (LF) can be arbitrarily specified (e.g. squared error, absolute error). At each stage a regression tree is fit on the negative gradient of the loss function. For the classic case where LF is the squared-error loss, the learning procedure entails consequentially optimizing the residual error.

To illustrate the algorithm, we refer to Figure 5. In this example, the data is fit using **five** decision tree regressors. Each panel in the top row illustrates the cumulative improvement due to the inclusion of the results from the succeeding steps. The residuals at each stage are plotted in the bottom row. The panels in Figure 5 are numerically labeled to illustrate the succession of steps relevant to fitting a gradient boosted decision tree to the example data.

The first step is to fit a weak decision tree regressor to the data, as shown in the first panel at the top (i.e Panel "1"). Notice that the model is quite too simple to capture the data. The residuals of this fit in Panel 1 is shown in Panel 2 (bottom left). A new decision tree is then fit onto the residuals in Panel 2. The sum of the decision trees in Panel 1 and Panel 2 are then added to create the decision tree regressor in Panel 3. The process is repeated three more times as decision trees are fit to the residuals in Panels 4, 6 and 8. The final model, shown in Panel 9, is then a (weighted) combination of all these previous learners.

Among the main advantages of using a GB regressor are as follows: (1) capable of working with data with features that are on different scales; (2) the capacity to support arbitrary loss functions, which it directly optimizes; and (3) that it is robust and often the best possible model. [1,2]

On the other hand, a major disadvantages include: (1) its sensitivity to noise and extreme values; (2) slow to train (but fast to predict); (3) and the several hyperparameters one needs to carefully tune to find the optimal model.[1,2]

The main model parameters which need fine-tuning are: (1) the number of estimators, $n_{estimators}$, (2) the tree depth, and (3) the learning rate, Ł. The number of trees, or weak learners, is determined by $n_{estimators}$. The tree depth is controlled by max_depth. The contribution of each learner to the final model is controlled by the learning rate (also called the shrinkage) parameter, Ł. Empirically, a small Ł (i.e. $Ł < 0.1$) leads to better generalization, compared to that in the absence of shrinkage ($Ł = 1$). Models with smaller learning rates, however, requires an larger number of estimators, leading to longer computational times.[3]

In Python, the gradient boosting regressor is implemented under the *sklearn.ensemble* package. Commercial web search engines as Yahoo and Yandex have used variants of gradient boosting in learning ranking functions for web search (Z. Zheng et al. 2007).

[1] M. Landry (2015)

[2] P. Prettenhofer (2015)

[3] S. Saita 2016

[4] Annotated figure grabbed from the GBT regression discussion video by A. Ihler.

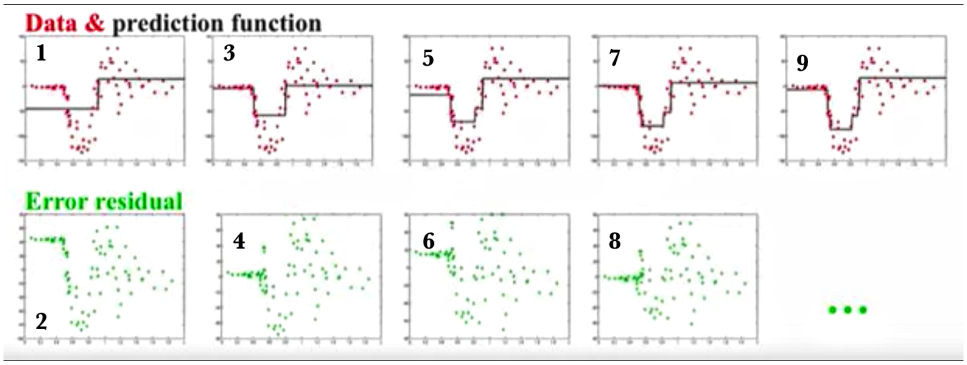

Figure 1: An illustration [4]of gradient boosting fitting. *Top:* Data (red points) and prediction function (black curve). *Bottom:* The error residuals shown as green points. *See text for discussion.*

4 Methodology

4.1 Statistical Properties of the Ames Housing Data

Figure 2 shows the $\log_{10}$ distribution of the sale prices of homes sold in Ames, IA from 2006-2010. The box plot representation of the sale prices is in the top panel, showing a few outliers both at the low and high price range. As depicted by the black curve overlaid on the histogram, the log of the sale price is non-gaussian, with a slight positive skew. From 2006-2010, the cheapest home sold for $34,900, while the fanciest home sold for $755,000. The average home is valued at $181,000. The median price is slightly less at $163,000.

In terms of the size of the properties, Figure 3 shows the distribution of the ($\log_{10}$) of the property size. An average property had a land area of 11,000 ft^2. The total land area of the properties ranged from 1300 - 215000 ft^2. For comparison, the most expensive property had a total area of 21,535 ft^2.

Along with continuous home variables, we also show the general behavior of some home features which are essential drivers of the home value. Such variables can take only discrete values, such as the number of full-size bathrooms, bedrooms, cars that fit in the garage, etc. Figure 4 show the sale prices vary as a function of these features. Here we see that the sale price is independent of the month the house was sold, but varies strongly as the total number of rooms and its overall condition grade. Further, the older the home the lower the price.

Figure 5 shows the general relationship between the sale price with some of the categorical features in the dataset. Here, different neighborhoods result to varying home values. As with intuition, homes with kitchens, garages, pools, fireplaces and heating in excellent condition demand higher prices. Land contour, shape and slope do not generally affect property values.

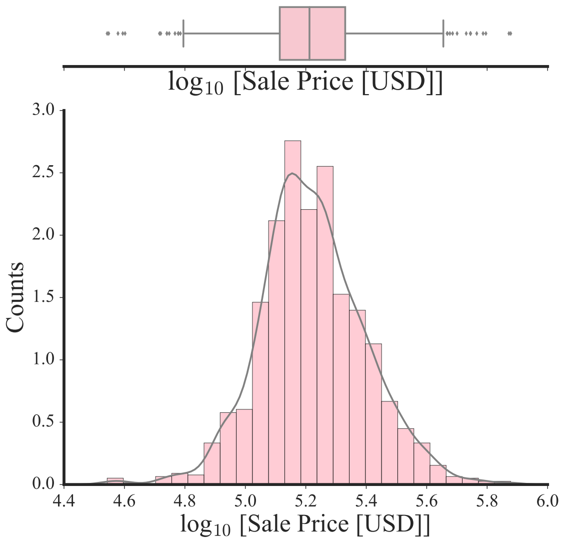

Figure 2: The distribution of the area of properties sold in Ames, IA from 2006 -2010 in the $\log_{10}$ - space. *Top:* A box plot representation of the lot areas, showing a number of outliers. *Bottom:* The distribution of the logarithm ($\log_{10}$) of the prices, overlaid with a negatively skewed non-normal fit to the distribution.

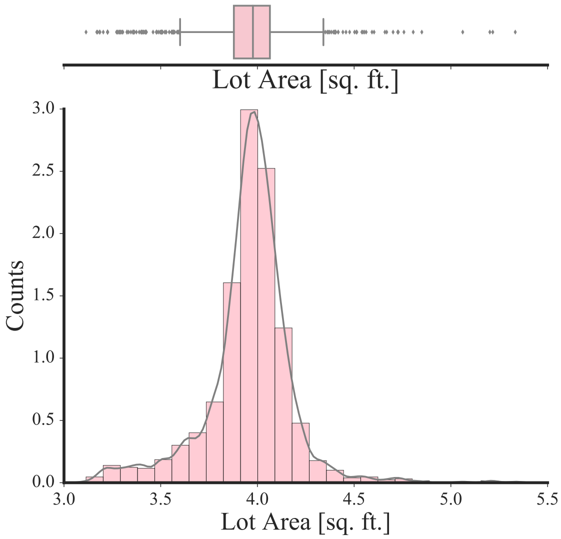

Figure 3: The distribution of the prices of homes sold in Ames, IA from 2006 -2010, shown in $\log_{10}$ - space. *Top:* A box plot representation of the prices, showing few outliers. *Bottom:* The distribution of the logarithm ($\log_{10}$) of the prices, overlaid with a non-normal fit to the distribution.

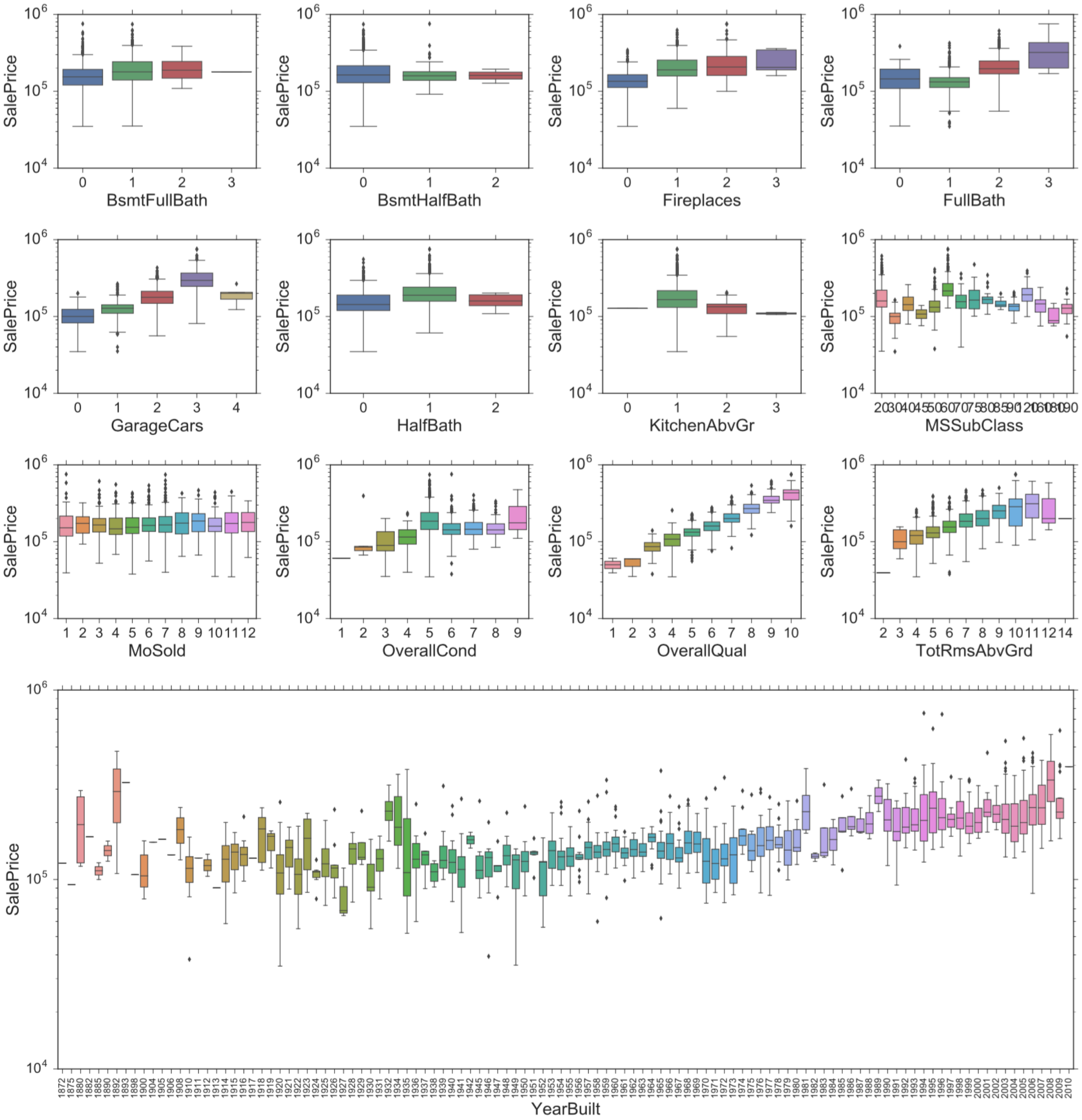

Figure 4: Visualizing correlations of discrete features in the Ames housing data with the sale price. The overall quality of a house strongly correlates with the sale price.

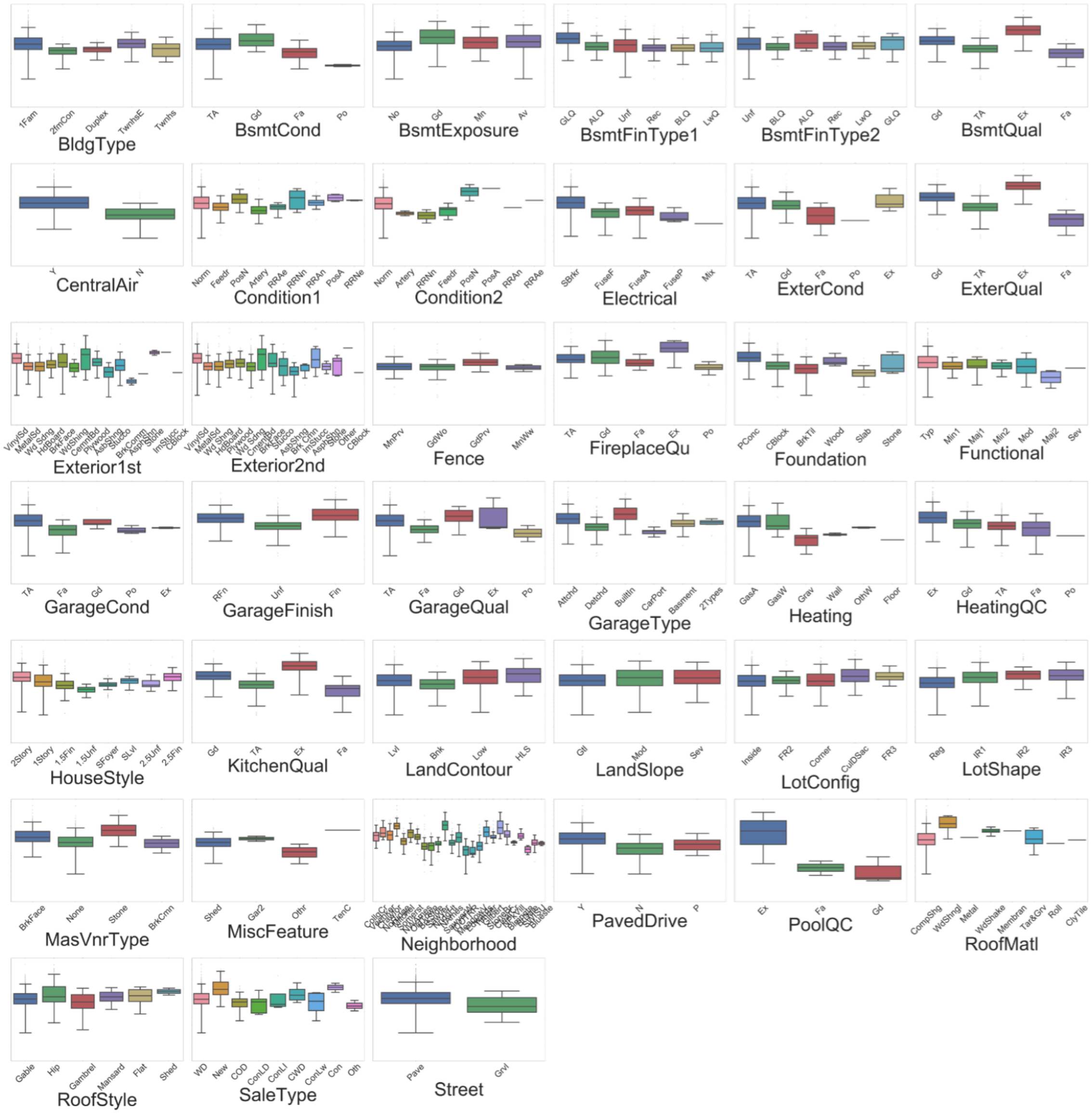

Figure 5: Visualizing correlations of categorical features in the Ames housing data with the sale price. The quality of garage, heating, and exterior of a house show some level of correlation with the sale price.

These exploratory results would be useful in determining which features and more important and more relevant relative to others. As many of these features are categorical, efficiently selecting which features to include in training would greatly help in reducing the dimensionality of the problem. The next section would discuss which features we would include in analysis and why.

4.2 Feature Selection

4.2.1 On Continuous Features

To be more discerning of a multiple regression analysis, it would be useful to examine the correlations between home features and the sale price. We start by examining how continuous features correlate with the sale price, and with all other continuous features. Figure 6 summarizes the Pearson correlation coefficients between any two continuous features. The red indicates a positive correlation, while blue corresponds to anticorrelation. The darker the hues, the stronger anti/correlations.

Most notably, Figure 6 tells us that the year the garage was built (GarageYrBlt) is strongly related to the year the house was built. This demonstrates that whenever a house was built, a garage was also built. The total basement area (TotalBsmtSF) stronly relates to the first floor area (1stFlrSF), implying that larger basements lead to more space for the first floor (and intuitively, vice-versa). Moreover, based Pearson coefficients with respect to SalePrice, we could exclude the following variables: BsmtFinSF2, LowQualFinSF, EnclosedPorch, 3SsnPorch, ScreenPorch, MiscVal and PoolArea.

4.2.2 Non-Continuous Features

With insights from Figures 4 & 5, it is apparent, at least qualitatively, that some features are less relevant than others. From Figure 4, the cost of the home is unlikely to be sensitive to the number of half-baths and kitchens, and the month the home was sold. From Figure 5, the neighborhood is a likely cost-driver, as well as the quality and condition of various sections of a house.

With insights from Figures 4-6, and after iterating through various combinations of relevant features, I find that the following feature set is the most concise and yields the best performance:

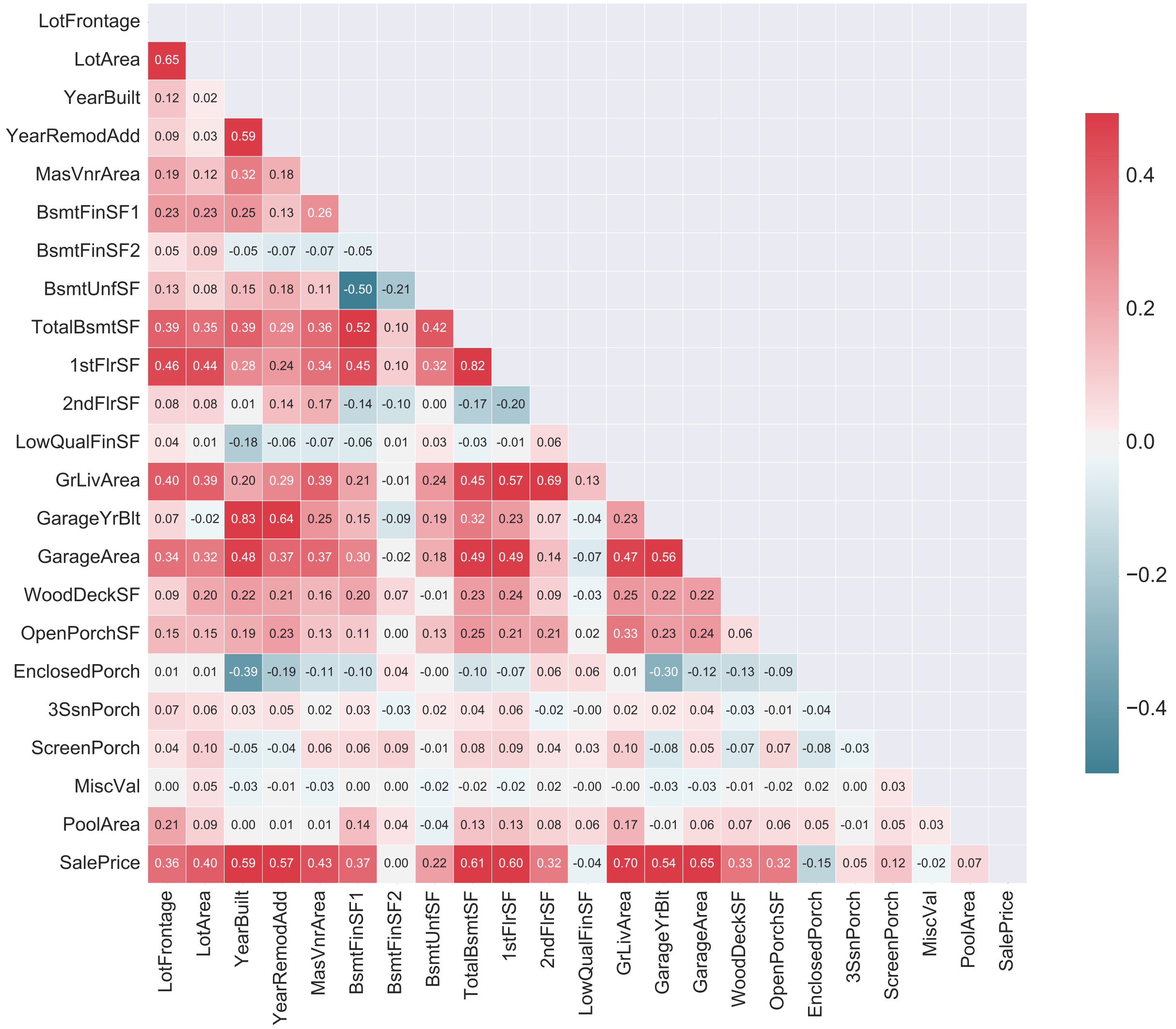

Figure 6: A heatmap of the Pearson correlation coefficients between pairs of features in the dataset. The redder (bluer) the hue, the stronger the correlations (anti-correlations). For this analysis, the SalePrice is the target. If the correlation coefficient between two non-target feature is large (i.e. $\Xi \geq 0.6$), then the feature that correlates more with the target variable is selected.

Table 1: Selected features for regression.

Regression Feature Set			
Continuous			
BsmtFinSF1	GarageArea	GrLivArea	LotArea
MasVnrArea	OpenPorchSF	SalePrice	TotalBsmtSF
WoodDeckSF	YearBuilt	YearRemodAdd	···
Discrete			
BedroomAbvGr	Fireplaces	FullBath	GarageCars
GarageCond	GarageQual	Heating	TotRmsAbvGrd
PavedDrive	PoolQC	SaleCondition	···
Categorical			
BsmtQual	Condition1	Foundation	ExterQual
HeatingQC	KitchenQual	Neighborhood	MSZoning
OverallCond	OverallQual	RoofStyle	···

4.3 Preprocessing

Thanks to Kaggle, little preprocessing of the data is involved for this analysis. As as this analysis is concerned, there were only two preprocessing steps involved. First, it is important to carefully understand what the NaN's actually signify for a given feature, before these examples be excluded. Second is the transformation of categorical features into dummy variables using **pandas.get_dummies()** function. For all columns in this dataset, an NaN indicates an *absence* of that component in a home. For instance, the absence of a pool would imply NaNs for both PoolQC and PoolArea. Hence, an NaN for a numerical column is replaced a the number **zero**. For categorical features, these are replaced by the code **"nothing"**, which now becomes an additional feature in itself.

4.4 Model Selection and Evaluation

The impressive advantages of using gradient boosted decision trees discussed in §3 led me to choose the GB regressor to solve the Ames Housing prediction problem. Note, however, that prior to the final choice of algorithm, two other ensemble regression methods were tested, namely, the random forests and Adaboosted decision trees. These resulted to a much poorer performance in terms of the R^2 and RMSE of the test set. §5 will tackle more of this.

Even with over 1400 examples at our disposal, it is imperative that we explore

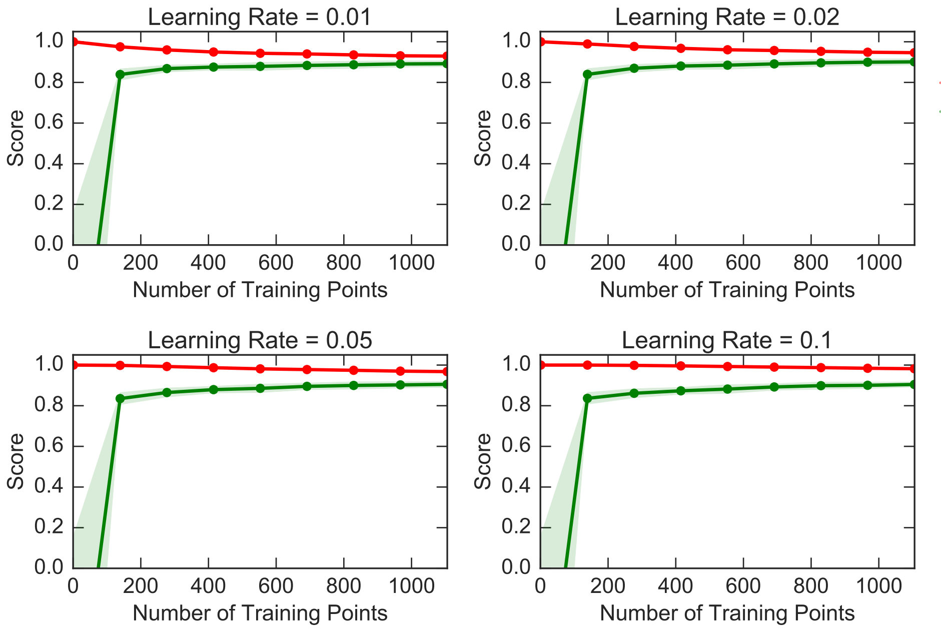

Figure 7: Learning curves as a function of the training size for various values of learning rates. The red curve is for training, while the green curve is for testing. At at least 1000 training points, a learning rate of Ł $=$ 0.02 gives the best performance. As the validation score still rises, albeit slowly, a larger training size could still prove useful these example values of Ł.

what the optimal training size would be for various parameters of the GB regressor. The learning curves of Figures 7 & 8 imply that given the feature set of Table 1, excellent performance will be achieved using about 1000 weak learners at a learning rate of 0.02, using roughly 1000 examples. Such set-up would require a train-test split of 80% for training, and the remaining 20% to be used for testing. We also note that the scoring metric is via the usual R^2 (coefficient of determination) regression score function.

Figure 9 zooms into the various learning rates using (max$_{depth}$, min$_{samples\_leaf}$, min$_{samples\_split}$, n$_{estimators}$) = (2, 3, 2, 1000). An Ł $=$ 0.02 gives the optimal model. At Ł $\gtrsim$ 0.02, the validation score saturates, and the model begins to overfit the data.

5 Results, Evaluation and Discussion

gfg

6 Conclusions

7 References

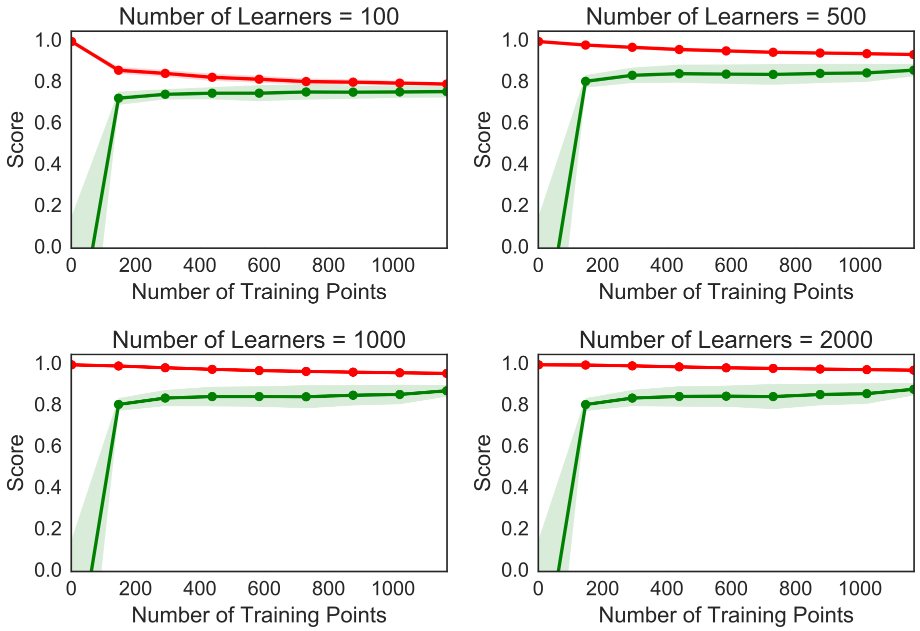

Figure 8: Learning curves for various values of $n_{estimators}$. For the parameter set ($\max_{depth}$, $\min_{samples\_leaf}$, $\min_{samples\_split}$, Ł) = (2, 3, 2, 0.1), the optimal model is given by $n_{estimators} = 1000$. (*Same color code as Figure 7.*)

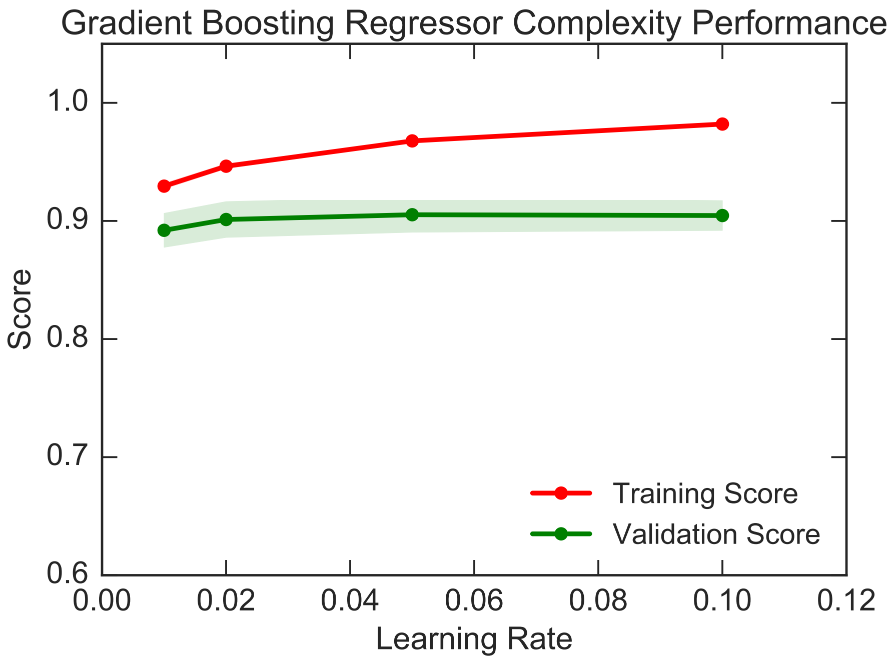

Figure 9: Learning curves for various values of learning rates.

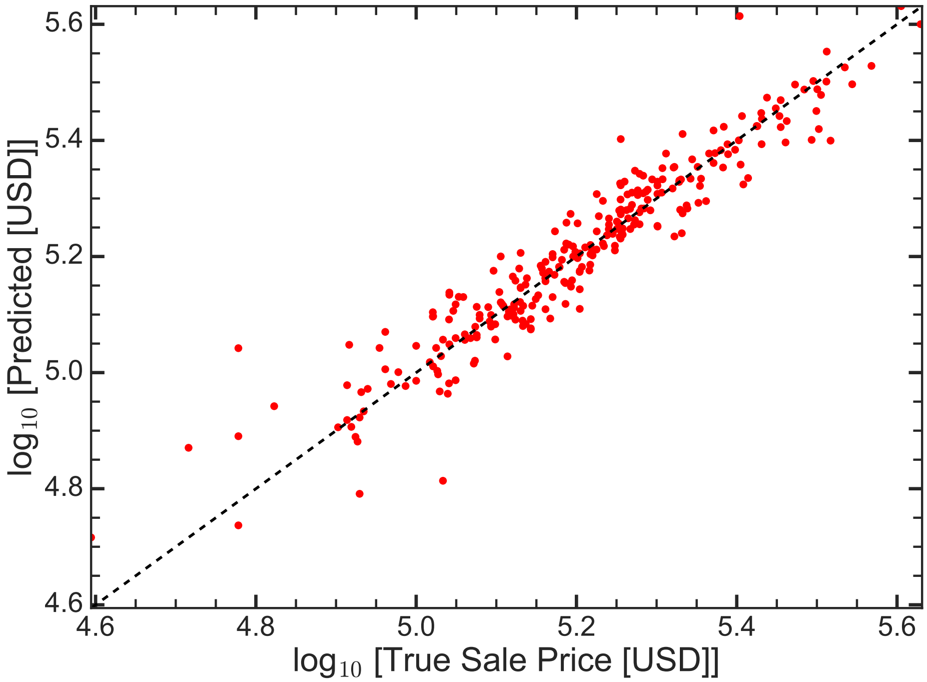

Figure 10: Comparison of the sale price predicted by the regression model to the **true** sale price, using the 276 training examples, derived from an 80−20 training-to-testing split. The black dashed line denotes the $y = x$ line, denoting a perfect fit.

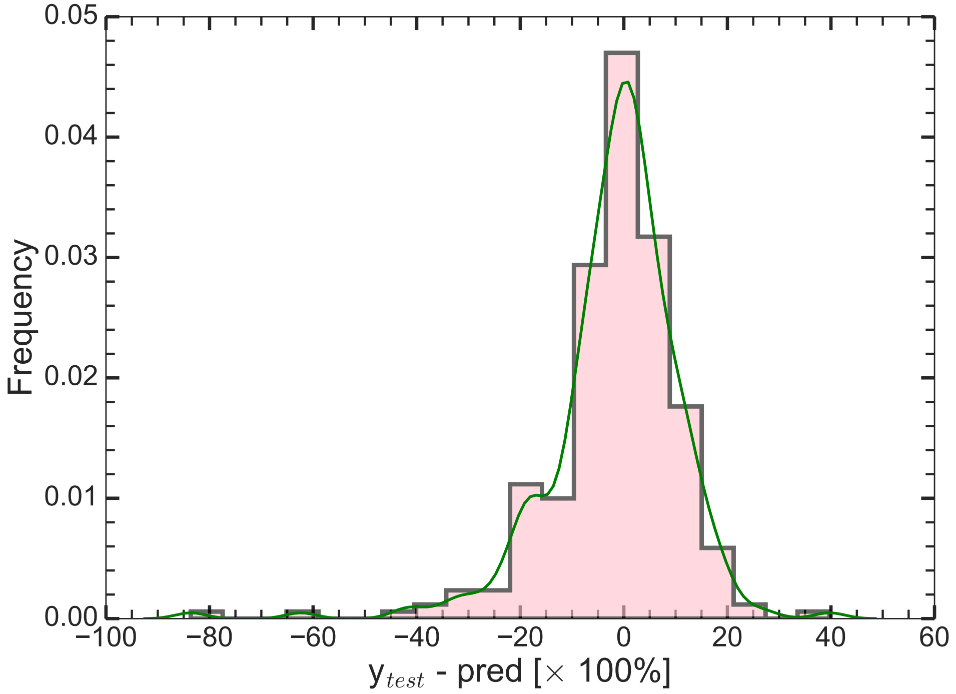

Figure 11: The fractional relative difference of the predicted price to the true sale price, overlain with a non-normal fit in green. The worst fit has a relative difference of 80% from the true sale price. The model has an $R^2 \sim 0.90$.

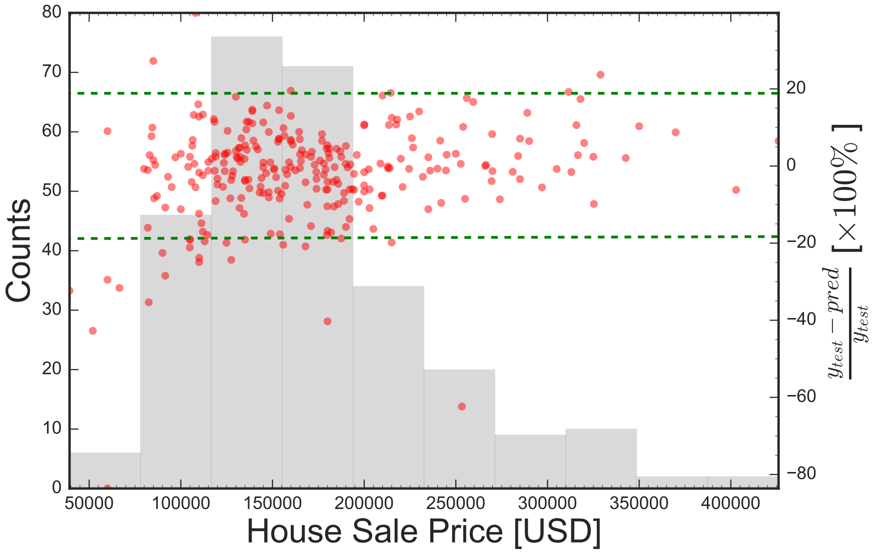

Figure 12: *Left axis:* The distribution of the sale prices in linear scale. *Right axis* The relative difference between the predicted and the true sale price. The green dashed lines denote the location of the $\pm 2\sigma$ range. The examples outside the green lines, which show the worst fit, have been over predicted by the model.

A Appendix

Table 2: Features of the Ames Housing dataset that were selected for this analysis.

Feature	**Description**	**Units**	**Selected**
LotArea	Lot size	square feet	Yes
LotFrontage	Linear feet of street connected to property	feet	No
GrLivArea	Above grade (ground) living area	square feet	Yes
TotalBsmtSF	Total basement area	square feet	Yes
BsmtFinSF1	Type 1 finished	square feet	Yes
BsmtFinSF2	Type 2 finished	square feet	No
BsmtUnfSF	Unfinished basement area	square feet	
LowQualFinSF	Low quality finish	square feet	No
1stFlrSF	First Floor Area	square feet	Yes
2ndFlrSF	Second floor area	square feet	Yes
MasVnrArea	Masonry veneer area	square feet	Yes
WoodDeckSF	Wood deck area	square feet	Yes
OpenPorchSF	Open Porch area	square feet	Yes
ScreenPorch	Screen porch area	square feet	Yes
EnclosedPorch	Enclosed porch area	square feet	No
3SsnPorch	Three season porch area	square feet	No
PoolArea	Pool area	square feet	No
MiscVal	Value of miscellaneous feature	USD	No
GarageArea	Size of garage in square feet	square feet	Yes
Discrete Features			
GarageYrBlt	Year garage was built	Year	Yes
YearRemodAdd	Remodel date	Year	Yes
GarageYrBlt	Year garage was built	Year	Yes
OverallQual	Rates the overall material and finish of the house	Numeric code	Yes
OverallCond	Rates the overall condition of the house	Integer code	Yes
YearBuilt	Original construction date	Year	Yes
BsmtFullBath	Basement full bathrooms	Integer	No
BsmtHalfBath	Basement half bathrooms	Integer	No
FullBath	Full bathrooms above grade	Integer	No

Continued on next page

Table 2 – *Continued from previous page*

Feature	**Description**	**Units**	**Selected**
HalfBath	Half baths above grade	Integer	No
Bedroom	Bedrooms above grade [5]	Integer	Yes
Kitchen	Kitchens above grade	Integer	No
Fireplaces	Number of fireplaces	Integer	No
MoSold	Month Sold (MM)	Integer $1-12$	No
YrSold	Year Sold (YYYY)	Year	No
GarageCars	Size of garage in car capacity	Interger	Yes
	Categorical Features		
MSSubClas	Identifies the type of dwelling involved in the sale	Integer code	No
MSZoning	Identifies the general zoning classification of the sale	Letter code	Yes
Street	Type of road access to property	Gravel or Paved	Yes
Alley	Type of alley access to property	Gravel or Paved access	No
LotShape	General shape of property	Shape code	No
LandContour	Flatness of the property	Letter code	No
Utilities	Type of utilities available	Letter code	No
LotConfig	Lot configuration	Letter code	No
LandSlope	Slope of property	Letter code	No
Neighborhood	Physical locations within Ames city limits	Letter code	Yes
Condition1	Proximity to various conditions	Letter code	Yes
Condition2	Proximity to various conditions [6]	Letter code	No
BldgType	Type of dwelling	Letter code	No
HouseStyle	Style of dwelling	Letter code	No
RoofStyle	Type of roof	Letter code	Yes
RoofMatl	Roof material	Letter code	No
Exterior1st	Exterior covering on house	Letter code	No
Exterior2nd	Exterior covering on house [7]	Letter code	No
MasVnrType	Masonry veneer type	Letter code	No
ExterQual	Evaluates the quality of the material on the exterior	Letter code	Yes
ExterCond	Evaluates the condition of the material on the exterior	Letter code	No
Foundation	Type of foundation	Letter code	Yes
BsmtQual	Evaluates the height of the basement	Letter code	Yes
BsmtCond	Evaluates the general condition of the basement	Letter code	No
BsmtExposure	Refers to walkout or garden level walls	Letter code	No
BsmtFinType1	Rating of basement finished area	Letter code	No
BsmtFinType2	Rating of basement finished area [8]	Letter code	No
Heating	Type of heating	Letter code	No
HeatingQC	Heating quality and condition	Letter code	Yes
CentralAir	Central air conditioning	Letter code	No
Electrical	Electrical system	Letter code	No
KitchenQual	Kitchen quality	Letter code	Yes
Functional	Home functionality [9]	Letter code	No
GarageType	Garage location	Letter code	No
FireplaceQu	Fireplace quality	Letter code	No
GarageFinish	Interior finish of the garage	Letter code	No
GarageQual	Garage quality	Letter code	No
GarageCond	Garage condition	Letter code	No
PavedDrive	Paved driveway	Letter code	No
PoolQC	Pool quality	Letter code	No
Fence	Fence quality	Letter code	No
MiscFeature	Misc. feature not covered in other categories	Letter code	No
SaleType	Type of sale	Letter code	No
SaleCondition	Condition of sale	Letter code	Yes

[5] Does not include basement bedrooms.

[6] If more than one is present.

[7] If more than one material exists.

[8] If multiple types exist.

[9] Assume typical unless deductions are warranted.