101C Project Report

Avash Monajemi, Yuqi Zhu, Jaehyeong Lee, Iris Liu

December 15, 2018

1 Introduction / Project Goal

The various quality and quantity of many physical attributes of a property provide new oppurtunities for prediciting the affordibility of a home in Iowa. There are 79 variables specifically which are directly related to property sale, and hence the affordability of a home. The challenge and goal is to clean the data in such a way that most robust classification techniques will yeild accurate predictions that result in the lowest misclassification rates.

1.1 Problem Statement

It is your job to predict the affordability for each house. For each Id in the test set, you must predict the value of the affordability variable. (Affordable vs. Unaffordable)

2 Data Preprocessing

Examining the dataset briefly gives preliminary insight for how the training and testing data should be manipulated which are : (1) Certain characteristics of a home have multiple variables linked to them, (2) Data collection from select predictors resulted in a number of NA results that ought to be corrected for, and (3) Existence of both numeric and character variables and how they may conflict during future data modeling procedures unless they are restructured or simplified. The following outlines the most common methods that are used to manipulate this data along with several examples and motivations for doing so.

2.1 Methods

Deletion of a Predictor Entirely

1. **Evidence claimed the predictor is not linked with affordibilitty**

- *Example* : Removal of the LotArea predictor due to evidence claiming that you cannot neccacarily determine the price of a home based on lot area.

2. **Existence of other predictors already explain the predictor**

- *Example* : BsmtFinType1 and BsmtFinType2 describe the rating of the basement finished area. After we take into account the existence of the predictor BsmtCond which evaluates the general condition of the basement, these variables now become obsolete by themselves as the general condition of the basement explains it enough.

3. **Predictor result is embedded in creation of new predictors**

- *Example* : FullBath and HalfBath variables are accounted for (totaled) in a new predictor which is total # of baths.

4. **Single level dominates the predictor (or too many NA's)**

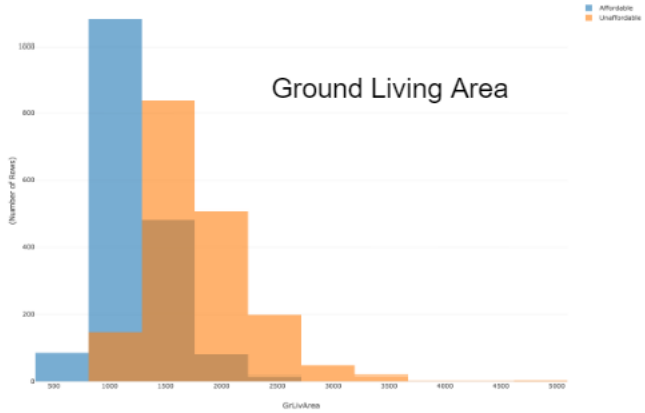
- *Example* : The majority of homes in the data do contain central air conditioning (CentralAir : Yes or No), so analysis would be limited with the inclusion of this predictor due to the homogeneity. This is deleted completely.

5. **Feature Creation**

- *Example* : We develop new variables from existing variables to extract hidden relationships of the data. As said earlier, we want to predict the affordabilitty of a home based upon number of baths and half baths. These individual variables may not have as strong of a connection with the response as the total number of baths so we uncover this to improve our future model accuracies.

Conversion of Numeric Variables to Factors(w/levels)

- *Example* : The Porch area in square feet (ScreenPorch) variable was highly skewed to the right (Figure 1), taking on many values of 0 and an extremely small number of positive square feet values. In light of this it is rather important to view this data as simply having a porch or not and convert to it a factor with corresponding levels.
- Note that we could also have transformed the Porch area variable (or other variables) by taking a log of this variable, correcting for some of the skewness. The variable of Ground Living Area is another example of a variable in which we may also take a logarithm of because of its skewness as shown below.



Simplification of Factor Levels

- *Example* : LotShape is simplified into a factor with only 2 levels. The motivation for this change is that there is no need to be too specific with the irregularity of a property shape and it is best classified as irregular or not.

Recoding of NAs

1. **Variable descriptions indicate NA is an absence of a feature**

- *Example* : Basement conditions with an NA indicates having no basement, and they are appropriately filled.

2. **NAs are filled by measures of central tendency / most commonly occuring class**

- *Example* : One observation for the training data's MSZoning (zoning classification of the sale) is an NA and therefore assigned to the most commonly occuring class of RL : Residental Low Density.

2.2 Importance of Data Preprocessing

Data preprocessing is crucial for machine learning methods to perform to the best of their ability. During the training phase of the model, if there are variables that contain (but not limited to), noisy data, outliers, unreliable predictors with too many levels, or missing values, these inconsistencies will in turn bring failure to our data mining techniques that follow. Consider if we did not remove any predictors, or by simplfying existing predictors by the methods recently explained. While we may undergo robust machine learning techniques such as random forest or decision trees and even tune parameters to use these methods in perfection, our data preprocessing methods will allow redundant variables (not linked to home affordability) to be included in our models which may higher our MSE and classification inaccuracies. In essence, we need to spend as much time cleaning our data as we do modeling.

3 Data Modeling

We consider a number of machine learning methods to model and predict the affordability of homes. After running on model on our test data, we observe the accuracy of our prediction and compare it to other learning methods and investigate why a method performed the way it did.

3.1 Methods Not Considered

We focus this assignment on algorithms that have a model fitting or training step. In other words, we will avoid methods that are "lazy learners" such as the Nearest-Neighbors method : they do not learn information from training data but instead memorize structure in the training data itself and use it for classification. There are disadvantages with these (often nonparametric methods) which include computation time for computing predictions as well as, for the Nearest-Neighbors method not included, a possible curse of dimensionality where in high dimensions, the method suffers due to no nearby neighbors to look for.

In addition, there are a few methods that although could have worked well for our data, time did not allow us to implement these or we did not have enough knowledge of these other methods.

3.2 Logistic Regression

We want to use Logistic Regression as our first method for classification and use it as a baseline that we will then compare with more complex and fancy algorithms. Furthermore, we want to utilize at least one method that will classify observations according to the *probability* that they correspond to a certain class. This means that if :

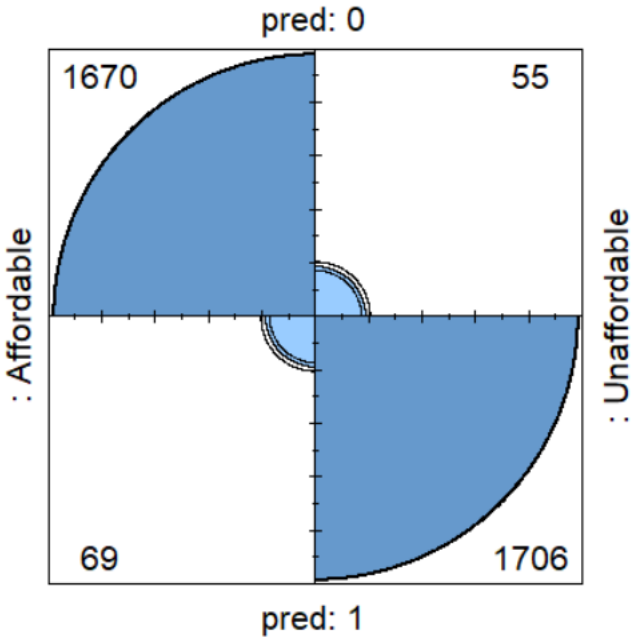
$$Pr(Affordability = Affordable|X) > 0.5$$

then we classify that home as being affordable!

3.2.1 Method / Results

We considered building the logistic regression model where the logistic model has a logit that is linear in X, estimating our coefficients of the model by the method of maximum likelihood, thereby obtaining our predictions.

The results our predictions can be seen by the confusion matrix below



We see that the confusion table does indicate there was some error in our prediction on the testing data. However, logistic regression gave an accuracy of 96%, which does set a tough baseline accuracy for future methods to come!

3.2.2 Limitations / Future Work

Although our accuracy of prediction was extremely high and we expect that some future methods may not surpass the 96% accuracy that logistic regression gave, there could be a reason why some error did occur.

- Logistic regression is prone to defficiencies with too many categorical predictors and our numerous conversions of numerical predictors to categoricals with levels may have deterriorated the accuracy of this model slightly.

Since logistic regression still produced a high accuracy, it may not have neccacarily been that the major assumption of a linear decision boundary was violated but rather our data preprocessing led to some inaccuracy.

3.3 Support Vector Machines

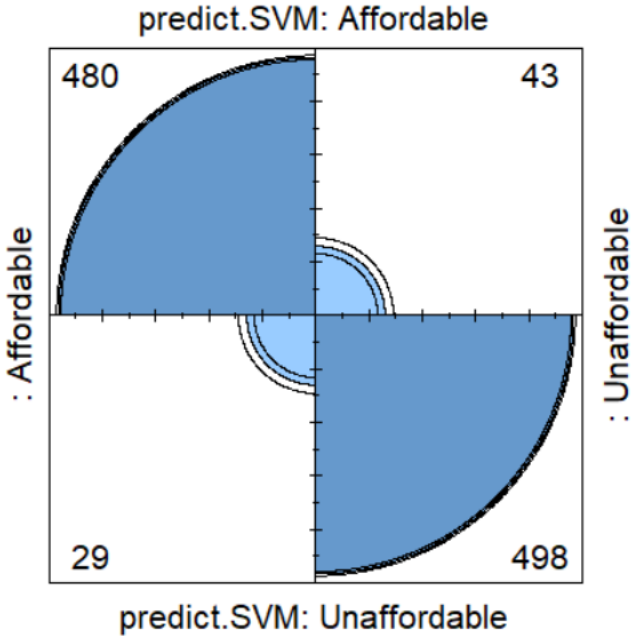
It is natural to consider Support Vector Machines as our next learning method as it has a relationship with logistic regression. Specifically, only the subset of observations, the support vectors, will play a role in the classifier and observations on the correct side of the margin have no effect (loss function of 0). Likewise, points that lie far from the decision boundary of logistic regression have *near* a loss of 0. Hence, these two methods go hand in hand and require some similar intuitions.

3.3.1 Method

The default SVM method was used which is of a non-linear (radial) kernal, in addition to a cost value of 1 and a gamma of 0.005050505. Our motivation was to use a non-linear kernal to correct problems that result with using linear kernals when actually non-linear class boundaries exist.

3.3.2 Results

By the confusion matrix results below, the method of SVM did result in a high accuracy as well (93%), but inferior to logistic regression by a tiny margin.



Because logistic regression outperformed SVM, the reason may be due to the seperation of classes. It is known that SVM brings better results when classes are not as well seperated. Therefore, it could be that the classes were very well seperated that logistic regression performed much better.

3.3.3 Limitations / Future Work

It is worth noting that SVM ought to have their parameters tuned for optimal performance. Though we chose a cost value of 1, the cost value should be chosen with cross validation, as a small value of cost yields a classifier that may have low bias but high variance, and a large value of C will bring higher bias but lower variance. Finding the best C is also important.

Likewise, because we chose a kernal of radial for our SVM method, we could have also tuned the gamma parameter because as gamma increases the fit is more flexible and non-linear which may be neccacary to improve accuracy.

3.4 Naive Bayes Classifier

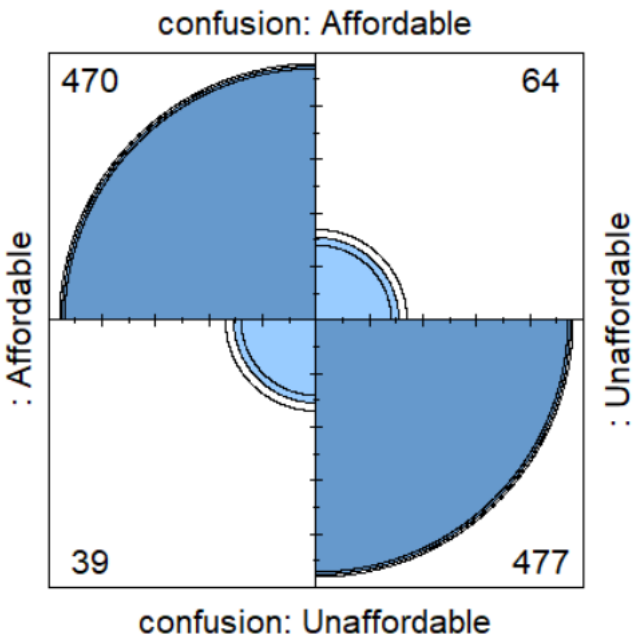
We chose Naive Bayes model because firstly, it is very easy and quick to use. Although there were many models at our disposal, since NB is not difficult to implement, we decided to give it a try. Secondly, NB can handle both discrete and continuous feature variables. Our housing data contained variables that belonged to both, so it is another reason we picked NB. Thirdly, NB is not sensitive to irrelevant features. This is one of the strong advantages that NB has because our housing data contained sheer number of up to 50 predictor variables. Upon observing each of them, we discovered that many variables may be deemed insignificant and choosing NB was great in mitigating this what could've been the main source of trouble.

· **Naive Bayes Assumption**

$$P(X_1, X_2|Y) = P(X_1|X_2, Y)P(X_2|Y) = P(X_1|Y)P(X_2|Y)$$

3.4.1 Limitations / Future Work

However, we should note some main disadvantages that NB inherently bears. NB makes very strong assumption on conditional independence among features. Hence the results can be potentially bad if the assumption is severely violated. Another problem is data scarcity. if you estimate the feature value using frequentist approach, you may obtain the result approaching nearly zero or one. This can lead to instable results and generate worse results.



With this model, we obtained prediction accuracy of 0.90, as shown above. This was fairly satisfactory results but relatively poor compared to those of other models. We suspect this due to the autocorrelation among feature variables.

3.5 Decision Trees

Decision tree was definitely one of our top choices for modeling. It is very intuitively clear and easy to interpret and lays out paths for different scenarios. It can also be combined with other techniques such as Support Vector Machines (SVM).

3.5.1 Method

We built a classification tree that contained 12 terminal nodes. We did not seek to prune the tree and check the results of a pruned tree. Although pruning and using cross-validation to find the optimal complexity of a tree is useful, we were knowledgeable of methods such as a random forest or boosting that already work to improve decision tree problems.

3.5.2 Limitations / Future Work

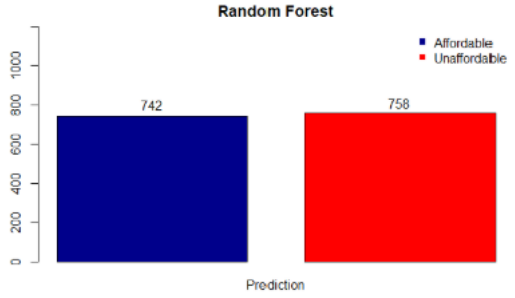
One of the notable shortcomings of decision tree is instability. Decision tree's structure is very vulnerable to just a little tweak within the data and hence can generate relatively inaccurate results. Another is that decision tree is biased in favor of features with more attributes and therefore it is important to tune down the number of levels before fitting decision tree onto the data.

With decision tree we got the prediction accuracy of 0.92, which was the second worst of all the outcomes. We think that the decision tree was too simplistic approach to take into account the complexity embedded within the relationship of housing price and other variables and is prone to overfitting.

3.6 Random Forest

3.6.1 Method / Results

Random Forest (RF) generated the best outcome out of 5 models. We divided the training data into new training dataset and testing dataset by 7:3 ratio. Then we used 8 as value for mtry parameter to fit RF model. As a result, we got the prediction accuracy of 0.98. We derive this high accuracy of random forest model from the fact that they normalize overfitting risk by averaging out trees of random sub-samples from the dataset. Hence this model resulted in less variance than other models and generated more accurate predictions. The actual predictions that this model assigned for the observations can be seen by counts below in a barchart.

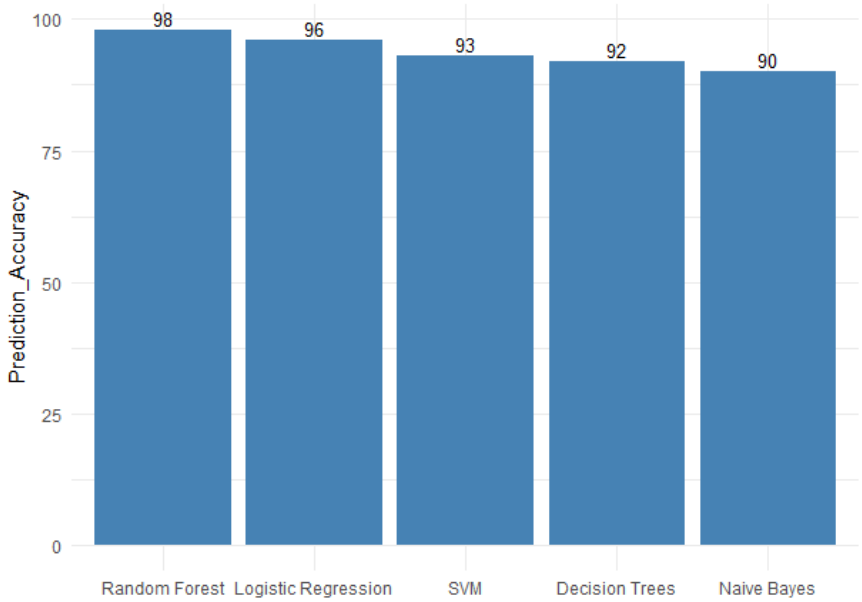


3.6.2 Limitations

Although RF performed very well, we should also note some of its caveats. Due to the complex nature of RF model, it's hard to interpret or visualize why it predicted certain housings as affordable or unaffordable. For example, if someone was curious in knowing why his or her property of choice was labeled unaffordable, it wouldn't be a trivial task to process the clear explanation that justifies the result.

4 Conclusion / Disclaimer

In conclusion, we have used 5 different learning methods that all brought us accuracies above or equal to 90%, and each method has their own advantages and disadvantages. Given our data cleaning methods and parameters chosen for each method, we see that the performance of the methods from best to worst is in the following order as shown by the barplot below : Random Forest, Logistic Regression, Support Vector Machine, Decision Trees, and lastly Naive Bayes



Lastly, though this competition was regarding selecting the method that gave the highest accuracy possible, all of the classifiers we have used performed well and can be argued to be useful by themselves in predicting home affordabilitty because they are better than guessing!

5 References

Drakos, G. (2018, August 12). Support Vector Machine vs Logistic Regression ? Towards Data Science. Retrieved from https://towardsdatascience.com/support-vector-machine-vs-logistic-regression-94cc2975433f

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). An introduction to statistical learning: With applications in R.

Tutorial on 5 Powerful R Packages used for imputing missing values. (2016, April 16). Retrieved from https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/