# Modeling Using Decision Trees

## Introduction

Decision tree methods have applications in both regression and classification analysis. Decision tree models are hierarchical methods that partition the space until the algorithm reaches a certain threshold. For this study, two decision tree methods, bagging and random forest, were used to train the predictive models.

Bagging and random forest make multiple passes on the data and results are aggregated from all the trees. Averaging reduces the issue of overfitting by canceling the uncommon features of data, yet producing a flexible fitting model. This often results in break-off with the bias-variance tradeoff. Before discussing these methods in details, we first give a brief introduction to bootstrap method.

Bootstrapping is used where the access to data is limited or sample size is too small to get reliable output. Bootstrap method randomly selects the sample data points with replacement to create a new subset of same size as the parent sample. Bootstrap method induces distortion in the data, which is fed as a new observation set to the machine-learning algorithm.

### Bagging

Bagging uses bootstrapped random data samples for growing the decision trees. All predictors are considered for the branch splitting and the terminal node observations are collected over multiple trees. The number of trees to grow are usually user defined based on how well the model performs on the cross validation or test data. Finally the terminal node observations from all trees are aggregated to produce the final output. Averaging over multiple trees reduces overfitting and the variance error. The one drawback of bagging is if few predictive features outperform the other, these features will be included in most of the trees. Thus the final averaging will be on correlated trees which does not reduce variance effectively despite using more trees. Random forests technique is used to improve on this limitation.

### Random Forest

Often times few predictors dominate at the branch split, because on average they perform better than their competitors. These “weak” predictors can be useful for local data feature but suppressed and rarely used. This results in most of the trees being correlated. Random forest adds one more step in the bagging algorithm. At the time of branch splitting, random forests only uses a subset of the predictor set. This is done over multiple trees, thus including almost all the features in the final results. This limits the error in bias and error in variance.

## Bagging for Predictive Analysis

Final output of the bagging model averaging over 500 trees is shown in Table 1. The mean of squared residuals is also called the Out-of-Bag (OOB) error. OOB error is computed by using only 2/3rd of the training data for training and other 1/3rd for validating the tree. Figure 1 plots the OOB against the number of trees. As the number of trees increased, the OOB error reduced almost exponentially. The rate of error reduction was really high in the beginning (from 1 to 40 trees). After 200 trees the magnitude of OOB error was almost constant, 0.13%.

Table 1 also lists the predictor importance summary and Figure 2 shows the corresponding plot. The term **%IncMSE,** is based upon the mean drop in prediction accuracy on the out-of-bag samples when the given predictor is not included in the model. **IncNodePurity** is the average decrease in the node impurity when split is over that particular variable. Among the 6 predictor time series, San Francisco County housing prices had relatively higher influence on Alameda County housing prices. From the set of other 5 predictors, time had relatively more influence followed by population.

Table2 compares the predictions and the test data. These magnitude of errors increased into the future months, indicating that bagging was more reliable for short term predictions (next one month or so). The test MSE was 29.51%.

Table 1 Bagging Model (R-output).

|  |
| --- |
| Bagging:: Model Summary |
| |  | | --- | | Call:  randomForest(formula = Alameda ~ .,data = AlamedaData,mtry = 11,importance = T,subset = train)  Type of random forest: regression  Number of trees: 500  No. of variables tried at each split: 11  Mean of squared residuals: 0.001347724  % Var explained: 99.93 | |
| Predictor Importance Summary |
| %IncMSE IncNodePurity  SantaClara 9.477318 20.95054501  ContraCosta 9.349447 20.77210016  SanFrancisco 13.076102 35.51751667  SanMateo 8.564825 15.81219437  Solano 8.072550 12.86226551  Marin 9.331034 15.11414058  time 10.058072 20.52681368  Pop 9.848822 18.35977401  CPI 9.430175 19.16544540  Inven 5.895297 0.09179841  TurnOver 4.591539 0.05488782 |

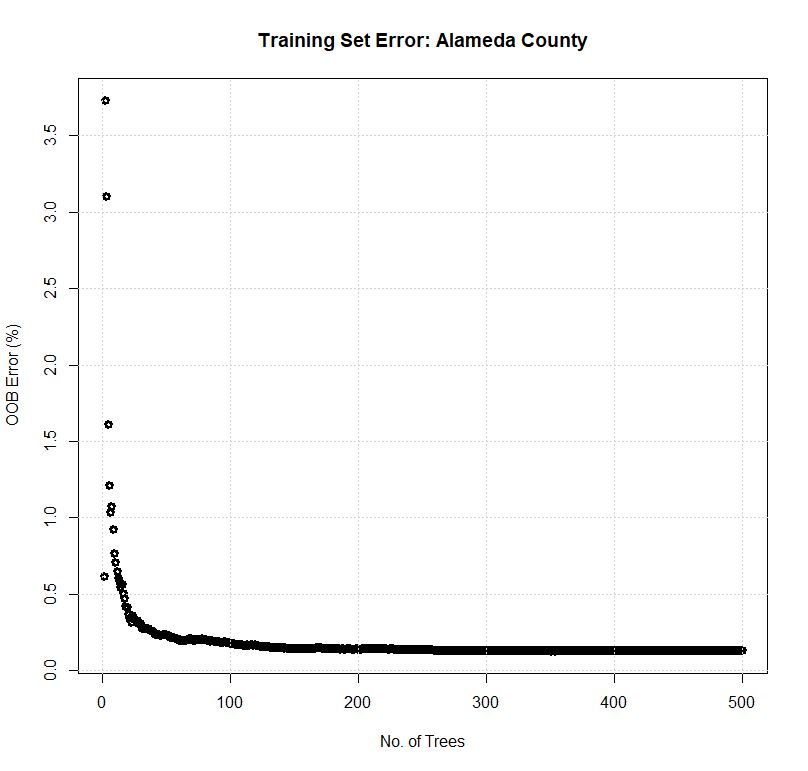


Figure 1 Bagging OOB error v/s no. of trees.

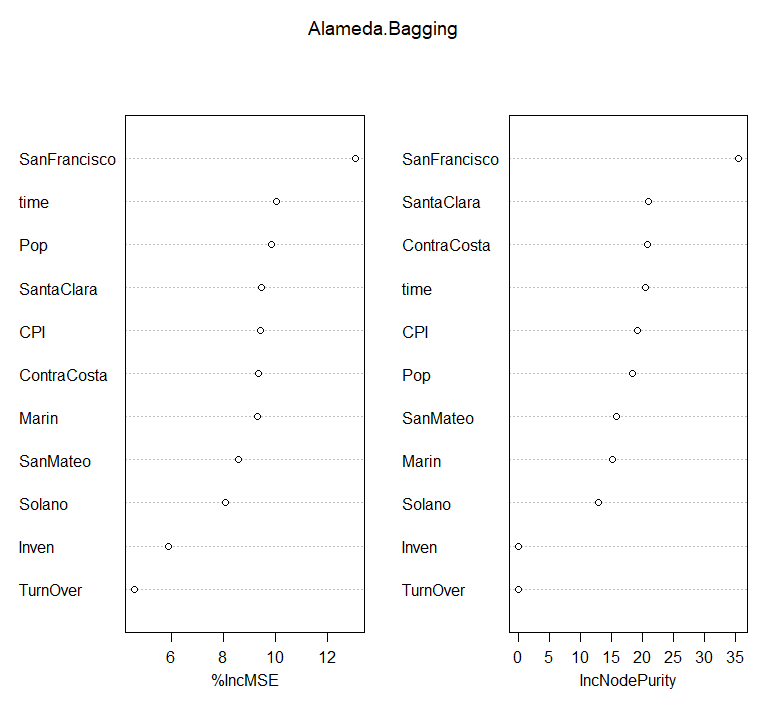


Figure 2 Predictor importance measures for bagging.

Table 2 Bagging Predictions and Test MSE

|  |
| --- |
| Test data, prediction and residuals |
| Test Data Model Predictions Residuals Residual^2  2018.01 8.423 8.137822 0.2851780 0.08132649  2018.02 8.569 8.135845 0.4331548 0.18762308  2018.03 8.654 8.132978 0.5210222 0.27146417  2018.04 8.698 8.126163 0.5718365 0.32699698  2018.05 8.755 8.100485 0.6545146 0.42838941  2018.06 8.786 8.096860 0.6891396 0.47491339 |
| Mean Test Error: **29.51%** |

## Random Forest for Predictive Analysis

Main difference between random forest and bagging is decorrelation of several trees on random bootstrapped samples. The idea is to reduce the variance by averaging the trees and eventually avoid overfitting by reducing the correlation between the random trees. Random forest uses a subset of predictors (without replacement) and thus multi-collinearity is not a major issue as all subsets are randomly selected. Table 3 gives the model summary of random forest. At each split a subset of 3 randomly selected predictors was used. The predictor importance summary suggested San Francisco County among the housing price predictors and population of the Alameda County from other 5 predictors to be relatively important, same as the bagging model (Table 1).

The OOB error of random forest and bagging was almost same (0.13%), Figure 3 compares the two. Random forest OOB error reduced faster than bagging during the initial increment in number of trees. After about 20 trees the random forest OOB error was half the magnitude of bagging OOB error. Thus on same training data set random forest performed better than bagging initially.

Table 4 compares the test data with the predictions from the random forest model. The mean test MSE of random forest model was 27.36% about 2% less than bagging model. Like bagging, random forest model predictions were more accurate for short term and the residuals increased further into the future.

Table 3 Random Forest Model (R-output).

|  |
| --- |
| Random Forest:: Model Summary |
| |  | | --- | | Call:  randomForest(formula = Alameda ~ ., data = AlamedaData, importance = T, subset = train)  Type of random forest: regression  Number of trees: 500  No. of variables tried at each split: 3  Mean of squared residuals: 0.001307916  % Var explained: 99.93 | |
| Predictor Importance Summary |
| %IncMSE IncNodePurity  SantaClara 11.316503 24.59108819  ContraCosta 9.639555 15.18653132  SanFrancisco 12.072966 26.77084846  SanMateo 10.171515 19.95339782  Solano 9.473884 15.83409830  Marin 10.251450 20.00878551  time 9.970050 19.15122676  Pop 9.896872 19.14841774  CPI 9.549034 18.73496533  Inven 3.625210 0.08620661  TurnOver 4.783974 0.04576179 |

Table 4 Random Forest Predictions and Test MSE

|  |
| --- |
| Test data, prediction and residuals |
| Test Data Model Predictions Residuals Residual^2  2018.01 8.423 8.146811 0.2761892 0.07628047  2018.02 8.569 8.146011 0.4229892 0.17891986  2018.03 8.654 8.144488 0.5095120 0.25960248  2018.04 8.698 8.138035 0.5599650 0.31356080  2018.05 8.755 8.133280 0.6217195 0.38653518  2018.06 8.786 8.132836 0.6531635 0.42662260 |
| Mean Test Error: **27.36%** |

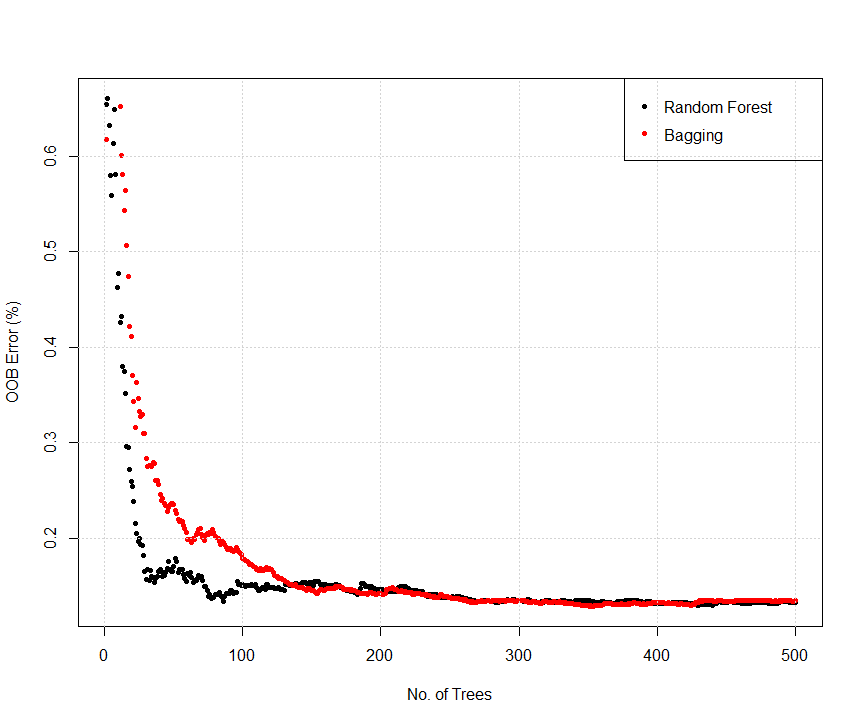


Figure 3 Comparison of Random Forest and Bagging OOB Error.

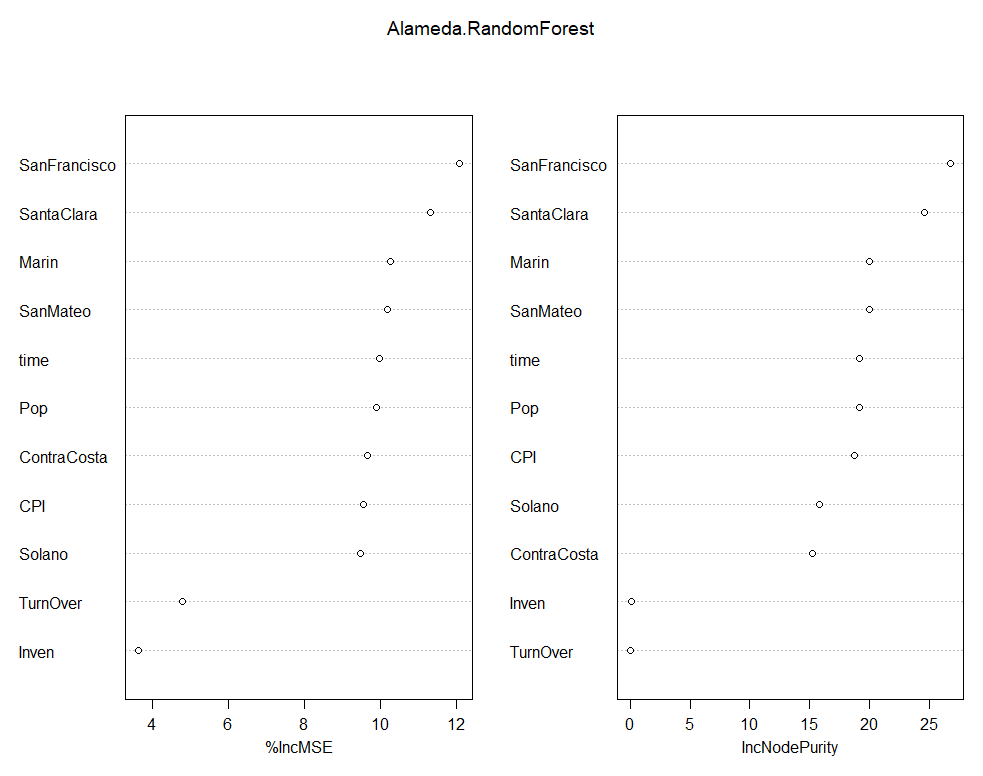


Figure 4 Predictor importance measures for random forest.

## Comparison of Bagging and Random Forest Models

Bagging and random forest models provided a good insight into the relative importance of predictor variables. San Francisco County had highest importance for Alameda County housing price models. Inventory and turnover percentage had least influence in the decision tree models. Performance of both models on training set was almost as the number of tree increased above 200, with OOB error of 0.13% after 500 trees. Both models suggested that San Francisco County was important predictor for Alameda County housing prices. The test MSE of random forest model (27.36%) was about 2% lower than bagging test MSE (29.51%). Thus the number of predictors chosen at the split did effect the test MSE. Figure 5 show the test and OOB errors plotted against the predictor subset size. Note that OOB error was scaled 100 times to compare its pattern with test MSE. The OOB error reduced significantly as the number of predictors increased from 1 to 2. At predictor subset size of 3, both OOB and test errors were at the minimum magnitude. After subset size of 7, the errors increased linearly. The last point on both curves corresponds to the bagging model with all 11 predictors. Figure 6 shows the squared-residual plots for both models plotted against each month. Random Forest performed slightly better than bagging in the months of May and June.

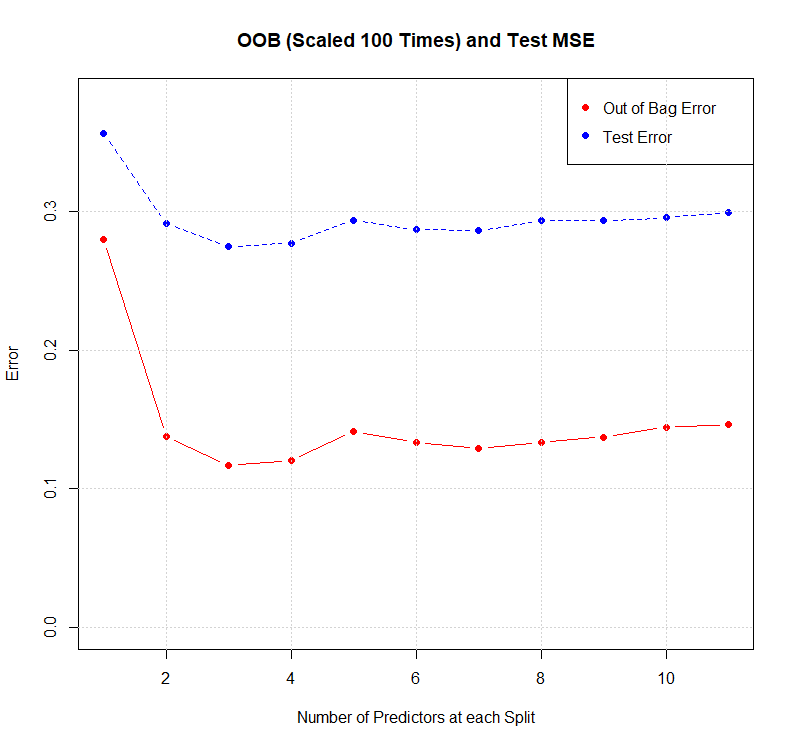


Figure 5 OOB comparison of Bagging and Random Forest.

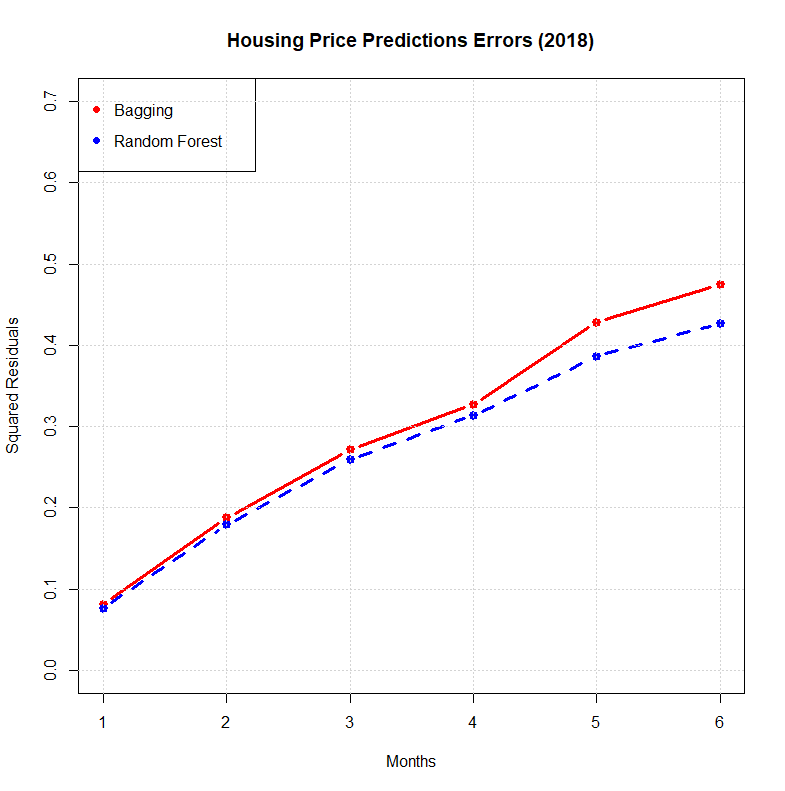


Figure 6 Squared residuals of housing Price prediction errors.