Roger Levaux, Joe Fregault

Stats 101C -- Final Project

Abstract

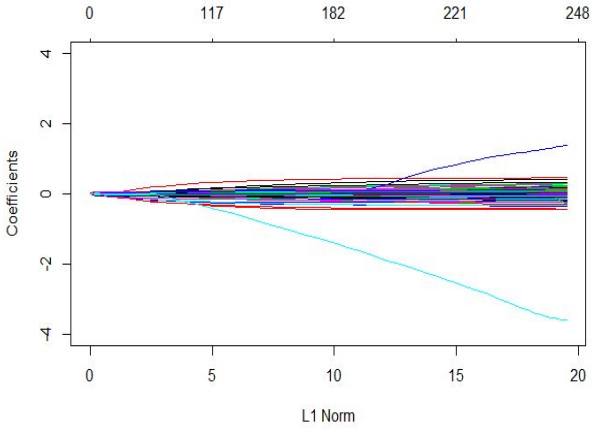
In this project, we were given a data set of basketball results from unknown teams and an unknown league with 9520 observations with 218 variables, and our goal was to predict whether or not the home team won. This process began with removing perfectly collinear variables. Following this, both linear and logistic regression were attempted before tree regression and a random forest model were settled upon. Our final model was a random forest model with the number of variables available to be split at each branch or mtry equal to 3, which achieved a kaggle rate of 0.6745.

Introduction

The home team winning is a 2 level factor (Yes or No question), thus this project consists of answering an inferential classification question. Given that we have the response available to us, we are using a supervised approach. Moreover, the coaches in the game have access to some data that we have, such as the 1st quarter score during their game's half time, and these variables would likely impact their decisions; as such, we cannot assume that the variables are independent. Independence can also not be assumed between the games themselves since the result of a game may affect a team's result going into the following game. This is in tandem with some variables being perfectly collinear. For example, variables such as HT.OTS and VT.TS, which both represent the visiting team's score, suggest that we cannot assume independence between either the features or observations nor can we assume that there is no significant collinearity between our variables.

After cleaning the data free of collinear variables, we first fitted linear regression and logistic regression models because these are natural initial choices for a classification question where the response variable can be dummied. Following these approaches, we applied a tree regression model and then concluded with a random forest model that produced satisfactory results.

Results



Model	Classification Rate	Kaggle Score
Linear Regression	0.6644737	0.62135
Logistic Regression	0.6588235	--
Tree Regression	0.654500	0.66019
Random Forests	0.6802632	0.6747

Discussions

To clean the data, we removed the variables that were perfectly collinear with one another and would serve no benefit to the model with their presence from the training data set. However, even though the variables were redundant, we found that random forest models that use all the variables outperformed the models which did not use these collinear variables. Moreover, as the graph of coefficients versus lasso regularization above on the left illustrates, even with minimal regularization, the majority of variables have their estimated coefficients clustered around zero, meaning that most variables are considered not very effective to predict the response.

We first fit a basic linear model, employing the variables that appeared to be the most statistically significant, which had a classification rate of 0.6645. Next, we implemented logistic regression, which was natural with a 2-level factor response, but only reached a test classification rate of 0.6589. Even picking our variables with stepwise selection to optimize our variable selection did not significantly improve our model. As such, we turned to tree regression because that type of model involves splitting our predictor space up before fitting regression within those splits. This immediately gave us a classification rate of 0.6545 -- already comparable to linear regression. Pruning the tree resulted in a lower classification rate, so next, we attempted a bagged random forest, which only had a rate of 0.64320. To improve upon this model, the main argument of concern was the mtry argument, which dictates the number of variables randomly sampled as candidates at each split, so flexibility increases as the argument's value decreases. Although the theoretical best value is the square root of the number of predictors, which in this case would be about 15, which indeed minimized MSE--when we dummied Yes/No to 1/0; however, we surprisingly achieved a top Kaggle score of 0.67475 with mtry = 3. This is likely because with so many predictors we needed our model to respond more to the data given to it, so this smaller value gave our model this property. Of note is that the square root of 15 is between 3-4, so it is possible with so many predictors, the square root of the square root of the number of predictors is the best option from an applied sense, although more research has to be done about this claim.

Conclusions

Despite the many models applied, we were still unable to achieve an extremely high classification rate; however, the rates achieved were acceptable. Random forests were nevertheless the best among the models we attempted. This was likely because decision trees are considered by some to mirror human decision making (ISLR pg 316), and basketball is, to vastly oversimply, a game of human-created strategy and human decision making in the heat of the moment. Thus, the likely reason that these random models describe the data well and are able to predict the outcome of a game is because they 'understand' the human decision making aspect in the background of the entire scenario than other models who do not have this property. Additionally, random forest models naturally feature select by the very nature of their algorithm, and with 218 features, this property is highly useful. Lastly, random forest models are also a good choice for this data because many of the variables are highly or perfectly collinear with each other, so the algorithm's method of randomly picking variables at splits decorrelates the variables in the process of predicting, which in turn betters the prediction by reducing variance.