Nicolas Bolanos Parra

Course Project

PREDICT 422 – Practical Machine Learning

Summer 2016

1. Introduction

A charitable organization wishes to develop a machine-learning model to improve cost-effectiveness of their direct marketing campaigns to previous donors. It has been observed from previous mailings that the average donation is $14.50 and the typical overall response rate is 10%. Each mailing costs $2.00 to produce and send, so it will be necessary to find a model that not only finds as many donor prospects as possible, but also is able to minimize unjustified costs.

The charitable organization is also interested in building a prediction model to predict expected gift (dollar) amounts from the mailing responses. This will be considered a separate model, which will predict the donation amount given that the mailing recipient has opted to donate during the current mailing campaign.

Such models are presented in this report, along with the reasoning process that lead to the prescription of such models. R code is provided as a separate file, but the algorithms and results are summarized in this report.

The entire data set consists of 3984 training observations, 2018 validation observations, and 2007 test observations. Weighted sampling has been used, over-representing the responders so that the training and validation samples have approximately equal numbers of donors and non-donors. The response rate in the test sample has the more typical 10% response rate. This over-representation should not affect the results, provided that the optimal mailing rate found from the model is adjusted before it is applied to the test data.

1.1 Methodology

The methodology of this analysis can be organized into three major steps:

1. Using the training data set, a total of 13 machine-learning methods were applied to find the optimal model for donor and donation amount prediction. From each machine learning method, a number of candidate models were developed, out of which one was selected as the preferred model.
2. Using the validation set, the 13 preferred models from each machine learning method were validated and compared, producing the two final models: one for predicting responses to the mailing campaign, and the other one predicting dollar donation amount for respondents.
3. Once the models have been selected, the test data set is used to make response and donation amount predictions, and to come up with an optimal mailing rate (after adjusting for over-sampling of donors).

Re-sampling methods were used during the first and second steps of the analysis to ensure the results obtained are generalizable and reproducible in the population. The models in the training data set were built, fine-tuned, and compared using 10-fold cross validation, since it is faster and less computationally intensive, while the bootstrap method was used to validate and compare the 13 models in step 2.

Model selection was primarily based on the total amount of profits calculated using the probability of class predictions (probability of donating or not donating) obtained from the classification model predictions (models 1-7 in the analysis) and on the Mean Squared Error (MSE) obtained from the donation dollar amounts obtained from the prediction models (models 8-13). In the case of the classification models, it is sought to maximize the following objective function:

Y(X) =14.50X – 2

(1.1.1)

Where Y is the cumulative sum of expected profits from the mailing campaign, X is a vector containing the probability of response of each mailing recipient, 14.50 is the expected donation amount of each donor, and 2 is the fixed cost of producing and sending the mailing.

By ordering the probabilities in vector X from highest to lowest, and calculating the cumulative sum for each additional mailing, it is possible to see how profits change. If the probabilities in X decrease below a certain level, the expected cumulative profits will begin to increase. It is the point before profits begin to decrease that we want to select as our cutoff probability value in order to discriminate donors from non-donors.

The purpose of such classification is to calculate the total profits using the following formula:

P(X) = 14.50(TP) - 2(FN + TP)

(1.1.2)

Where P(X) is the total profits for the campaign, TP is the number of true positives (predicted donors that, in fact, donated) and FN is the number of false negatives (predicted non-donors that, in fact, donated). In other words, the criterion for selecting the classification model is a function of the model’s ability to find true positives, and its ability to avoid the prediction of false negatives.

The MSE criterion, on the other hand, seeks to find the model whose mean prediction error is minimum.

2. Analysis

2.1 Exploratory Data Analysis

Before starting the modeling process, it is usually helpful to conduct data exploration. Familiarization with the data set helps the analyst make better modeling decisions and develop strategies to mitigate problems found in the data. One of the problems found in this data set is that the variables have different scales, which can be a problem in many machine-learning methods since variables with larger magnitudes can have large effects and the results. For that reason, it is necessary to standardize the data so that every continuous variable has mean of 0 and standard deviation of 1. The drawback to this solution is that the resulting models may be more difficult to interpret, since there are now variables with negative values that should otherwise be positive.

Another aspect of the data that may potentially create problems in the model building process is that some variables are highly skewed. Some machine learning methods may struggle with these variables, so it is necessary to also consider different transformations that can normalize the distribution of these variables. The following figure shows some of the variables that needed to be transformed:

|  |  |
| --- | --- |
| Raw Variable | Transformed Variable |
|  |  |
|  |  |

Figure 2.1.1

Another problem found in this data set is that many of the predictor variables are highly correlated (see Figure 2.1.2). Collinearity refers to the situation in which 2 or more predictive variables are closely related to each other. The presence of collinearity can pose problems in the regression context, since it can be difficult to separate out the individual effects of collinear variables on the response (James, Witten, Hastie and Tibshirani, 2013). As a result, the parameter estimations of regression models tend to be highly variable with minor changes in the training set, and can therefore lead to inflated errors in prediction. Mitigating the effects of collinearity amounts to carefully selecting the predictor variables in each modeling scenario. One helpful metric of the collinearity introduced by a predictor variable is the variance inflation factor, which will be used to make decisions on variable selection during model building, whenever possible.

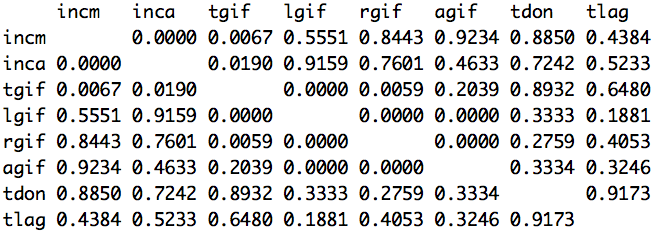


Figure 2.1.2

Other aspects of this data will become apparent during the model building process and the details will be provided as they arise.

2.2 Classification Models

A total of 7 classifications modeling techniques were used to develop the candidate models. The following sections explain the implementation of each modeling technique. The performance results are also shown here.

2.2.1 Model 1: Decision Trees

Decision trees have many different purposes. They can be used for both classification and regression problems. However, the main reason why this technique was employed first is because it can also be used to understand relationships between the variables, thanks to the fact that they are easy to interpret and they have a nice graphical representation. They also have the advantage of having variable selection embedded into their algorithm. Therefore, it can be used to see which variables are important for prediction.

When creating decision trees the analyst can decide the amount of pruning, which essentially controls the number of nodes and branches in the tree. In practical terms, the less pruning there is the higher the fit to the provided data. Of course, over-fitting a tree can lead to serious prediction problems due to the excess of variance. But on the other hand, excessively pruning the tree can also lead to problems in prediction due to excess in bias. Hence, the amount of pruning must be carefully selected based on the model’s ability to make out-of-sample predictions.

Thanks to re-sampling techniques such as 10-fold cross validation, tuning the amount of pruning is a matter of repeatedly fitting the model to a sub-sample, making predictions on the holdout set, calculating the performance metric in 1.1.2 and then averaging those metrics over the total number of iterations. Such is the model evaluation procedure employed in each classification modeling technique, including decision trees.

Based on that procedure, the following results were obtained for each of the pruning value.

|  |  |  |
| --- | --- | --- |
| Model | Pruning parameter | Cross-validation calculated profits. |
| Model 1.1 | 2 | 1342.45 |
| Model 1.2 | 3 | 2045.70 |
| Model 1.3 | 4 | 2154.45 |
| Model 1.4 | 5 | 2147.20 |
| Model 1.5 | 6 | 2147.20 |
| Model 1.6 | 7 | 2174.75 |
| Model 1.7 | 8 | 2202.30 |
| Model 1.8 | 9 | 2270.45 |

Table 2.2.1.1

It is observed that Model 1.8 (pruning parameter = 9) results in the highest cross-validation calculated profits, which is an indication that this model will perform better in out-of-sample predictions.

The following figure shows a diagram of Model 1.8

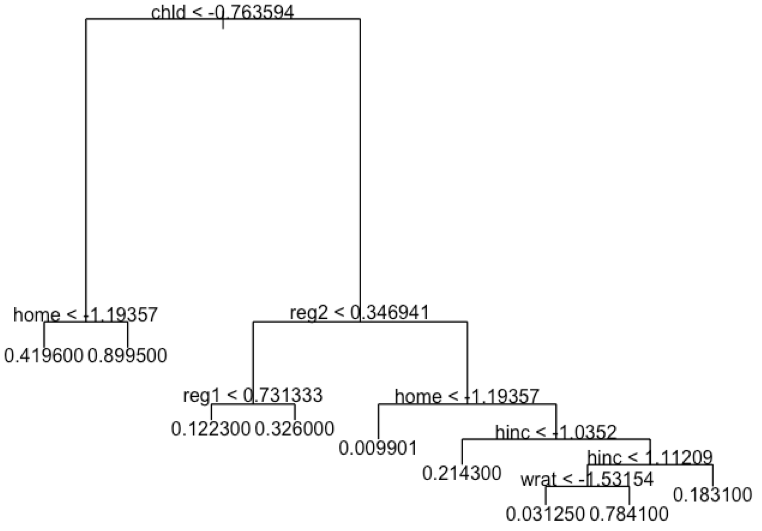


Figure 2.2.1.1

The tree diagram shows that the variable chld is the most important predictor in this classification problem. Other important variables are reg2, home and hinc.

2.2.2 Model 2: Random Forests

Even though the selection of the pruning parameter of the decision tree in Model 1 attempts to balance the variance-bias trade off, it is likely that attempting to fit a decision tree using a different sample will yield different results. Decision trees suffer from high variance for this reason (James, et al., 2013).

To avoid this problem, random forests generate bootstrap samples from the training set, fit a regression tree in each sample and makes predictions using each tree. The resulting predictions are then averaged, effectively reducing prediction variance. Furthermore, random forests randomize the number of predictors available for each bootstrap sample fit, which essentially decorrelates the trees and therefore contributes in the reduction of variance. This could be helpful in this case to discard the possibility of seeing similar tree structures in each sample.

On top of the re-sampling method embedded in the random tree’s algorithm, and in order to fine-tune the algorithm’s mtry parameter (number of variables used in each bootstrap sample fit), the 10-fold cross validation procedure was applied. The results are shown below:

|  |  |  |
| --- | --- | --- |
| Model | mtry parameter | Cross-validation  calculated profits |
| Model 2.1 | 5 | 2200.85 |
| Model 2.2 | 6 | 2197.95 |
| Model 2.3 | 7 | 2202.30 |
| Model 2.4 | 8 | 2193.60 |
| Model 2.5 | 9 | 2179.10 |
| Model 2.6 | 10 | 2186.35 |

Table 2.2.2.1

The results indicate that Model 2.3 has the best performance and mtry=7 is selected as the optimal parameter value.

2.2.3 Model 3: Boosting

Boosting is another decision tree-based approach to modeling whereby small-sized trees are fit sequentially in an attempt to explain the remaining unexplained variance in the data set. In other words, a tree is first fit to predict the outcome, and then another tree is fit using the residuals from the previous tree, which is then shrunken and added to the current prediction function. The procedure continues until no more significant variance can be explained, or until the maximum number of iterations is reached.

Boosting has three tuning parameters: the number of trees, which has been constrained to 5000 for this analysis, the shrinkage parameter, which indicates how fast (or slow) the boosting algorithm learns in each iteration, and the complexity of the tree fit in each iteration. The results are shown below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Response Distribution | Number of Trees | Complexity of Trees | Shrinkage | Cross-validation  calculated profits |
| Model 3.1 | bernoulli | 5000 | 2 | 0.001 | 2466.20 |
| Model 3.2 | bernoulli | 5000 | 3 | 0.001 | 2460.40 |
| Model 3.3 | bernoulli | 5000 | 4 | 0.001 | 2469.10 |
| Model 3.4 | bernoulli | 5000 | 2 | 0.200 | 2461.85 |
| Model 3.5 | bernoulli | 5000 | 3 | 0.200 | 2463.30 |

Table 2.2.3.1

2.2.4 Model 4: Linear Discriminant Analysis (LDA)

LDA attempts to estimate the decision boundary of the Bayes classifier by making a number of statistical assumptions and estimations. It assumes that each class sample comes from a population with a normal probability function, that there is a shared variance among the class distributions. It also has to estimate the prior probabilities of each class, and the mean and variance of their distributions (James, et al., 2013). Application of LDA also assumes that the decision boundary is linear, which may result in large biases depending on the modeling situation. One way of testing whether these assumptions hold for this data set is to use cross validation to obtain, compare and examine prediction performance.

Model 4.2 in this analysis was built with the aid of a R package called subselect. The subselect package addresses the issue of variable selection in a variety of statistical contexts, including LDA. This package assumes that all potential variable subsets can be ranked according to a well-defined numerical criterion and that the ultimate goal is to select the best subsets for any given cardinality (Cadima, Cerdeira, Silva and Minhoto, 2012).

Following the methodology of this analysis, multiple LDA models of different sizes were created and 10-fold cross validation was applied to obtain predicted profit estimations of the mailing campaign. The following results were obtained.

|  |  |  |
| --- | --- | --- |
| Model | Cross-validation  calculated profits | Variable selection method |
| Model 4.1 | 2479.25 | Full model |
| Model 4.2 | 2479.25 | Subselect package |
| Model 4.3 | 2470.55 | According to variable importance in decision tree |

Table 2.2.4.1

According to the cross-validation results Models 4.1 and 4.2 outperformed 4.3. However, Model 4.2 is preferred because it is less complex (i.e., has less variables) than the full model.

2.2.5 Model 5: Quadratic Discriminant Analysis (QDA)

Like LDA, the QDA classifier results from assuming that the observations from each class are drawn from a Gaussian distribution. However, QDA also assumes that each class has its own covariance matrix, which results in a non-linear decision boundary. Once again, to verify whether the assumptions and approximations of QDA apply to this data set, 10-fold cross validation is applied using the training set (and bootstrap in the validation set). The cross validation results are shown next:

|  |  |  |
| --- | --- | --- |
| Model | Cross-validation calculated profits | Variable selection method |
| Model 5.1 | 2483.60 | Full Model |
| Model 5.2 | 2482.15 | Using the subselect package for LDA |
| Model 5.3 | 2415.45 | Using variables found in decision tree |
| Model 5.4 | 2483.60 | Using the subselect package for logistic regression |
| Model 5.5 | 2483.60 | Using the subselect package for logistic regression |
| Model 5.6 | 2483.60 | Using the subselect package for logistic regression |
| Model 5.7 | 2483.60 | Using the subselect package for logistic regression |

Table 2.2.5.1

According to the results Models 5.1, 5.4, .5.5, 5.6 and 5.7 perform equally well. However, judging by the model complexity, Model 5.4 is preferred.

2.2.6 Model 6: Logistic Regression

Logistic regression approaches the classification problem by finding parameter estimates in the logistic function using maximum likelihood estimation. Finding the right model amounts to finding the right set of predictor variables, which is done here by using the subselect package. Five models were found, Model 6.1 being the full model and models 6.2-6.5 being the best subset models of cardinalities 10-13. The results are shown next.

|  |  |  |
| --- | --- | --- |
| Model | Cross-validation  Calculated profits | Variable selection method |
| Model 6.1 | 2454.60 | Full Model |
| Model 6.2 | 2463.30 | Subselect, cardinality = 10 |
| Model 6.3 | 2461.85 | Subselect, cardinality = 11 |
| Model 6.4 | 2457.50 | Subselect, cardinality = 12 |
| Model 6.5 | 2456.05 | Subselect, cardinality = 13 |

Table 2.2.6.1

It was found that the Model 6.2 with cardinality 10 performed better.

2.2.7 Model 7: K-Nearest Neighbors (K-NN)

The K-NN algorithm takes as it’s only parameter the number of neighbors it should use to make predictions. It is important to be careful not to select this parameter too low because it will lead to over-fitting and increase the variance of the model. On the other hand, selecting too many neighbors can also cause issues because it may lead to excessive bias in the model. The following plot shows how the cross-validation profit calculation changes as the parameter increases.

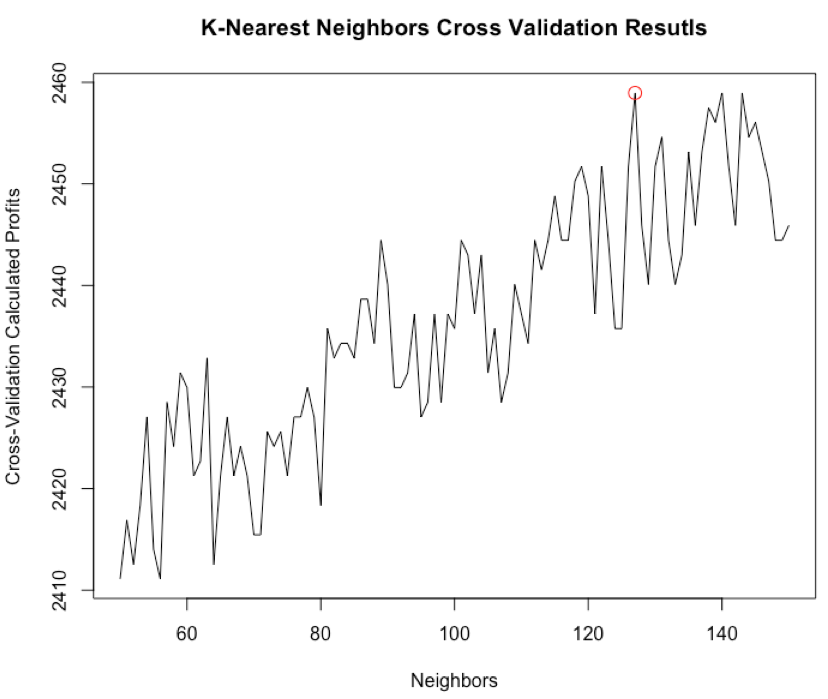


Figure 2.2.7.1

The trend is for the cross-validation calculated profits to increase as the neighbors parameter increases. However, there is substantial variation at each value of the neighbors parameter, which makes it doubtful that this model will consistently perform as observed in this case. Nevertheless, it tends to stabilize when the number of neighbors is 127 (marked in red in Figure 2.2.7.1), which is why the model with the neighbors parameter equal to 127 is selected.

2.3 Prediction Models

2.3.1 Model 8: Regression Trees

Decision trees can also be applied to regression problems. Although they do not usually perform as well as other prediction models, they are useful for finding significant predictors and understanding the relative importance of each predictor variable in the modeling scenario.

For this model, two subsets of predictor variables were provided. In the first case, Model 8.1 only the raw variables were used and in the second case, Model 8.2 the raw variables and the transformed variables were used. In both models, it was found through cross validation (cv.tree in R) that the optimal number of nodes in the tree is 10. Applying 10-fold cross-validation, it was further found that both models performed equally. They were, in fact, the same model. The results are shown below.

|  |  |
| --- | --- |
| Model | Cross-validation  MSE |
| Model 8.1 | 2.044899 |
| Model 8.2 | 2.044899 |

Table 2.3.1.1

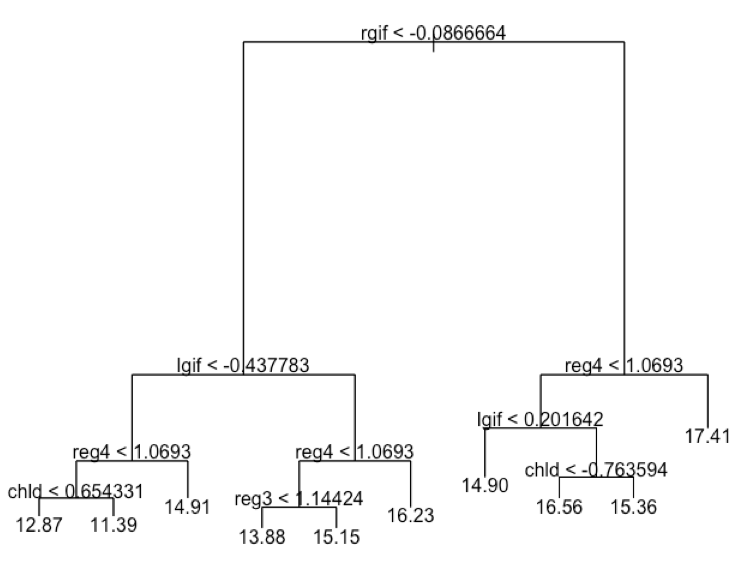


Figure 2.3.1.1

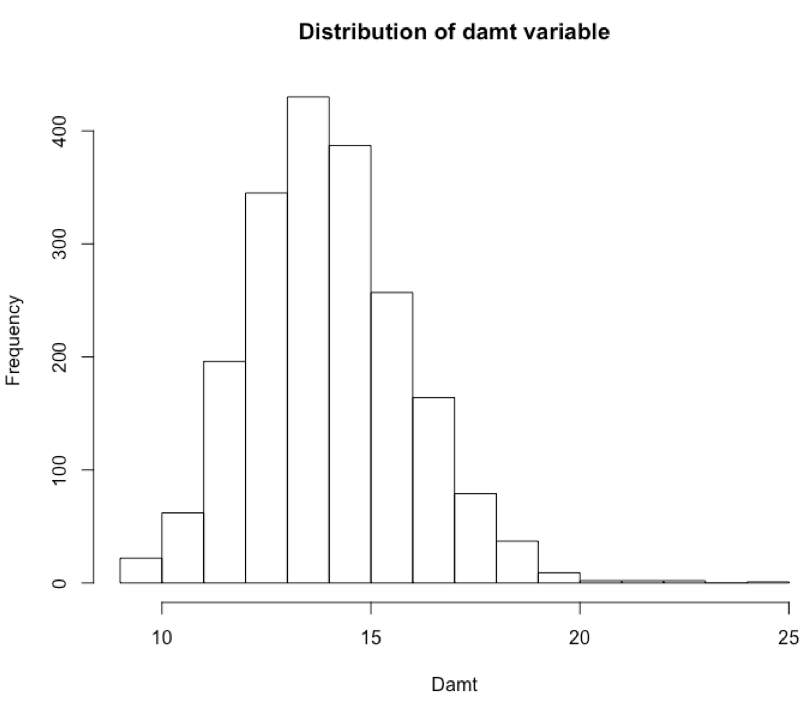
The model diagram shows that the variable rgif seems to have high predictive value. The variable reg4 and lgif seem to be very important as well.

2.3.2 Model 9: Ridge Regression

Ridge regression attempts to find a different and more flexible approach in balancing the bias-variance trade-off than linear regression. It uses a parameter λ to regulate the amount of bias and variance, which lends itself to using re-sampling methods, such as cross-validation, to find the optimal value of λ. The glmnet package in R contains a function cv.glmnet that can find this value for a given model.

Another improvement over linear regression is that ridge regression, while is not capable of selecting the optimal set of variables, it can reduce the effects of unimportant variables, minimizing the risk of over-fitting the model.

One question that came to mind when modeling the damt variable (donation amount) was to whether approximate it to a (skewed) Gaussian distribution or to a Poisson distribution, since it consists of positive integer numbers. The distribution of the damt variable is shown in Figure 2.3.2.1.



Based on that inquiry two models were fit. The first model (Model 9.1) specified Poisson as the response variable distribution and the second model assumed a Gaussian distribution (Model 9.2), both using the optimal value of λ found using the glmnet package in R. From the results shown below, it is likely that the Gaussian distribution approximates the response variable better.

|  |  |  |
| --- | --- | --- |
| Model | λ | Cross-validation MSE |
| Model 9.1 | 0.2207023 | 1.386301 |
| Model 9.2 | 0.126294 | 1.377597 |

Table 2.3.2.1

2.3.3 Model 10: The Lasso

The motivation to use the Lasso is similar to that of Ridge Regression. The difference is that the Lasso is truly capable of selecting the predictor variables, reducing those that are irrelevant to have no impact on the response variable estimations. Again, this is regulated by the parameter λ and which is also selected using the cross-validation function of the glmnet package in R. In this case, two λ values were used: the one found to have the least cross validation error, and another value yielding an error that is one standard error away from the optimal. Since the resulting cross-validation MSE was significantly higher for the second λ value the first model (Model 10.1) was preferred.

|  |  |  |
| --- | --- | --- |
| Model | λ | Cross-validation MSE |
| Model 10.1 | 0.007571124 | 1.369195 |
| Model 10.2 | 0.1124288 | 1.518502 |

Table 2.3.3.1

2.3.4 Model 11: Generalized Additive Models (GAM)

One of the common issues of building models with multiple predictor variables is that it is difficult to visualize the direct relationship between all predictors and the response variable. However, with the GAM it is possible to examine every pair-wise relationship between each predictor and the response variable, produce a model for each relationship and then add those models together into a generalized additive model.

The candidate models were built based on visualizing scatterplots of each predictor variable and the response. If there seemed to be little to no shared variance, the variables were left out of the models. If the relationship did not seem linear, either the predictor variable was transformed to linearize the relationship, a cubic spline was used to model the non-linear relationship, or both treatments were applied. An example is shown in Figure 2.3.4.1.

|  |  |
| --- | --- |
|  |  |
| Figure 2.3.4.1 – Left: example of a cubic spline applied to model the agif-damt relationship. Right: example of fitting a model after transforming the predictor variable. | |

The AIC criterion was also used to aid in the selection of variables and basis functions. The results of the 10-fold cross validation are shown in the following table.

|  |  |  |
| --- | --- | --- |
| Model | Cross-validation MSE | Cross-validation AIC |
| Model 11.1 | 2.358014 | 6636.708 |
| Model 11.2 | 2.390891 | 6662.187 |
| Model 11.3 | 1.988786 | 6326.889 |
| Model 11.4 | 1.519128 | 5844.453 |
| Model 11.5 | 1.514052 | 5839.522 |
| Model 11.6 | 1.512961 | 5837.950 |
| Model 11.7 | 1.513287 | 5836.696 |
| Model 11.8 | 1.405116 | 5709.189 |
| Model 11.9 | 1.324305 | 5593.727 |

Table 2.3.4.1

Model 11.9 seems to perform best, having both the least MSE and AIC.

2.3.5 Model 12: Linear Regression

It was previously mentioned that linear regression may encounter problems when there is multicollinearity. Another issue stems from the fact that the response variable is not quite normally distributed, so it may be possible that the linear regression assumptions of normality of errors is violated. Nevertheless, different variable transformations were tried and different selection procedures were used, including forward, backward, stepwise selection procedures, the subselect selection functions and some trial and error. To keep the models from having multicollinearity problems, the VIF criterion was used, and variables were kept or removed accordingly (usually if sqrt(VIF) > 2). The results are shown below

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Selection Method/Cardinality | Cross-validation AIC | Cross-validation MSE |
| Model 12.1 | forward | 565.5078 | 1.372432 |
| Model 12.2 | backward | 549.8216 | 1.371267 |
| Model 12.3 | both | 549.7746 | 1.372673 |
| Model 12.4 | 8 | 672.6379 | 1.496311 |
| Model 12.5 | 9 | 630.7700 | 1.453629 |
| Model 12.6 | 10 | 590.1913 | 1.392712 |
| Model 12.7 | 11 | 574.4368 | 1.393049 |
| Model 12.8 | 12 | 563.7209 | 1.375857 |
| Model 12.9 | 13 | 556.4909 | 1.369321 |
| Model 12.10 | 14 | 553.0293 | 1.374198 |
| Model 12.11 | 15 | 550.9578 | 1.370206 |
| Model 12.12 | Selected Manually | 618.2887 | 1.415675 |
| Model 12.13 | Selected Manually | 668.6781 | 1.455443 |

Table 2.3.5.1

Although Model 12.9 showed the least MSE and one of the lowest AIC, further examination of the coefficients indicated that some were not significant. Also, significant multicollinearity was found. Model 12.6 was found to perform best without suffering of excessive multicollinearity, while still showing one of the lowest AIC.

2.3.6 Model 13: Principal Components Regression (PCR)

Dimension reduction via principal components is a convenient way of simplifying the regression problem, especially when there are highly correlated variables in the dataset (see Figure 2.1.2). The intention is to find principal components to summarize the data and eliminate redundancies and then use those components to build a linear regression model. The question is how many components should we retain? Following the methodology of this analysis, the number of components has been selected based on the 10-fold cross validation performance. The number of components is based on the raw variables and does not include any variable transformations. The following table shows the results.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Model | Number of components | Cross-validation MSE | | Model 13.1 | 1 | 3.750555 | | Model 13.2 | 2 | 2.713785 | | Model 13.3 | 3 | 2.681052 | | Model 13.4 | 4 | 2.597513 | | Model 13.5 | 5 | 2.182591 | | Model 13.6 | 6 | 2.137410 | | Model 13.7 | 7 | 2.086242 | | Model 13.8 | 8 | 2.048850 | | Model 13.9 | 9 | 2.035330 | | Model 13.10 | 10 | 2.011769 | | Model 13.11 | 11 | 1.858109 | | Model 13.12 | 12 | 1.862013 | | Model 13.13 | 13 | 1.751263 | | Model 13.14 | 14 | 1.689775 | | Model 13.15 | 15 | 1.692004 | | Model 13.16 | 16 | 1.693280 | | Model 13.17 | 17 | 1.690344 | | Model 13.18 | 18 | 1.693866 | | Model 13.19 | 19 | 1.646251 | | Model 13.20 | 20 | 1.645559 |   Table 2.3.6.1 | Figure 2.3.6.1 |

Not much dimension reduction is possible without sacrificing performance. Other models have seen to perform better. With this in mind Model 13.5 is preferred because it compromises complexity and performance.

2.4 Validation Results

In spite of the fact that cross-validation was used specifically to validate the performance of each model, it is still necessary to perform further checks to avoid selecting a model that performs poorly on a different sample. Validation for this analysis was performed only on the 13 best-performing models of each machine-learning technique, and it consists of taking 1000 bootstrap samples from the validation set, fitting the model and estimating the response variable. To measure performance, the MSE is calculated on each of these bootstrap samples, which are then averaged to come up with an estimation of what the MSE would be on the population (or in another sample from the same population). The results of this procedure for the classification problem are summarized in the following table.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Bootstrap approximation of mailing profits | Bias | Standard Error |
| Model 1 | 11632 | -14.002 | 510.8235 |
| Model 2 | 12487.5 | 5.3 | 283.8391 |
| Model 3 | 12328 | 60.2895 | 292.1754 |
| Model 4 | 12357 | -35.736 | 314.0321 |
| Model 5 | 11994.5 | 48.6895 | 297.165 |
| Model 6 | 12226.5 | 26.685 | 310.7822 |
| Model 7 | 6415 | -344.541 | 223.4474 |

Table 2.4.1

Statistically speaking, assuming the sampled estimated mailing profit distributions are normally distributed, Model 7 is the only model that performs significantly worse than the others. However, based on the bias reported from the bootstrap estimations and the reported mailing profits, Model 2 is preferred. Model 2 was built using random forests.

The following table shows the results obtained for the prediction problem using a similar procedure.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Bootstrap approximation of MSE | Bias | Standard Error |
| Model8 | 2.07776 | -0.00710705 | 0.1548316 |
| Model9 | 1.487828 | -0.04405707 | 0.1510042 |
| Model10 | 1.474355 | -0.04835028 | 0.1507193 |
| Model11 | 1.247932 | -0.09445278 | 0.1162653 |
| Model12 | 1.459581 | -0.04697706 | 0.1504346 |
| Model13 | 140.7616 | -0.032573 | 1.616515 |

Table 2.4.2

Anova tests indicate that the only models performing significantly worse are models 8 and 13. Even though Models 9, 10, 11 and 12 perform fairly similar, Model 11, which is built on the Generalized Additive Model approach, is selected for prediction.

2.5 Prediction Results

Fitting Model 2 to the entire training data set and predicting mailing responses on the Test set suggests that the organization should mail to the 291 recipients with the highest donation posterior probabilities. This represents a total fixed cost of $582 for the mailing campaign.

Calculating the total profits depends on the expected donation amount, which requires using Model 11 to predict the donation amount for each recipient. Averaging the results indicate that for this particular mailing list, the expected donation amount for each recipient will be of $13.78, which is lower than the historic average.

3. Conclusion

A variety of machine-learning techniques were applied to the mailing classification problem and the donation amount prediction problem. However, this approach is not idiosyncratic of this particular analysis; it is good practice to apply different modeling techniques to any modeling scenario: Each modeling technique makes different statistical assumptions about the data or has been developed with a specific modeling problem in mind, which may not be known a priori. It is also helpful to have a reference point before deciding which model performs sufficiently well.

Similarly, proper modeling requires re-sampling techniques in order to arrive at reliable, generalizable and reproducible results. Drawing multiple sub-samples from a population sample is effectively like drawing multiple different samples from the population (Chaturvedi, 2016), which is crucial to building robust models that will hold for the population.

Based on these considerations, and those presented throughout this paper, there is great confidence that the proposed models will help the charitable organization improve cost-effectiveness of their direct marketing campaigns to previous donors.

Reference

Cadima, Jorge, et al. "The subselect R package." (2012).

Chaturvedi, Anil. PRED-422 Sync Session 6. 28 July 2016. Lecture.

James, Gareth, et al. *An introduction to statistical learning*. Vol. 6. New York: springer, 2013.