# Final Project

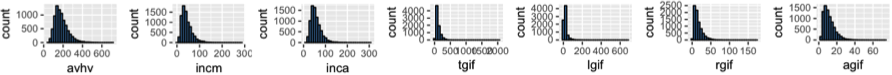
## Introduction

In this report we develop machine-learning models to optimize a charitable organization’s direct marketing campaign. Our goal is two-fold: to identify likely donors using a classification model in order to maximize the net profit from the mailings, and to estimate the gift amounts from donors using a prediction model. We created several classification models, including logistic regression, logistic regression general additive model (GAM), linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), k-nearest neighbors, decision tree, boosted trees, random forests, and support vector machines, in order to determine which model best identifies likely donors. We also created several prediction models, including least squares regression, best subset selection with k-fold cross-validation, principal components regression, partial least squares, ridge regression, and lasso regression in order to predict the amount of donations.

## Data Exploration

The data we are using for this report is the charity data set. This data set contains 3984 training observations, 2018 validation observations, and 2007 test observations. The data has been weighted in order to make the training and validation samples each contain roughly an equivalent number of donors and non-donors. The charity data set includes two response variables, donr (donor) and damt (donation amount), for the classification models and prediction models, respectively. It also contains 20 predictor variables, including the geographic regions of previous donors, whether the donor is a homeowner or not, the number of children the donor has, the donor’s income, the donor’s gender, the amounts of previous donations, and various other wealth, income, and donation factors that may help create strong classification and prediction machine-learning models.

Prior to developing any models, we examined our data in order to see if there were any missing values or abnormalities in the data. Although the data set did not contain any missing values, we noticed that several of the variables might benefit from transformations. We decided to apply logarithmic transformations to seven right-skewed variables: avhv, incm, inca, tgif, lgif, rgif, and agif. The top row of **Figure 1** shows the variables prior to transformation, while the bottom row shows the variables after transformation.[[1]](#footnote-1) As the figure shows, the logarithmic transformation helped normalize the distributions of these seven variables.



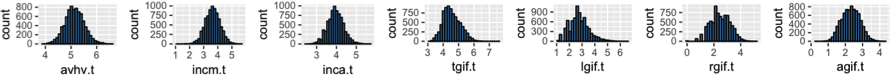


Figure : Transformation of Variables

After transforming our variables, we created a correlation matrix to gain further insight into the data. **Figure 2** shows a visualization of the correlation matrix, while **Table 1** shows the values of the most statistically significant correlations.

|  |  |  |
| --- | --- | --- |
| var1 | var2 | corr |
| damt | donr | 0.9817018 |
| tgif | npro | 0.8734276 |
| rgif | lgif | 0.8512241 |
| inca | avhv | 0.8484572 |
| inca | incm | 0.8296747 |
| agif | lgif | 0.8294224 |
| plow | incm | -0.8120381 |
| agif | rgif | 0.7706645 |
| plow | inca | -0.7510141 |
| incm | avhv | 0.7304313 |
| plow | avhv | -0.7187952 |
| damt | chld | -0.5531045 |
| donr | chld | -0.5326077 |



Figure : Correlation Matrix Table 1: Significant Correlations

As **Figure 2** and **Table 1** reveal, certain variables have strong correlations. For instance, tgif (dollar amount of lifetime gifts to date) has a very strong positive correlation with npro (lifetime number of promotions received to date). Some variables, such as plow (percent categorized as “low income” in potential donor's neighborhood) and incm (median family income in potential donor's neighborhood in $ thousands), have a strong negative correlation. The correlations are something we bear in mind as we create our classification and prediction models.

## Classification Models and Analysis

# Model Technique 1: Logistic Regression

The first classification technique employed is logistic regression, a simple and robust model for two response classes that is easily interpretable and applicable. The first of four logistic models includes all independent variables, the second model utilizes only those variables which are significant at p=0.05, the third model includes all independent variables, but also incorporates additional quadratic terms for numerical variables. Backward subset selection performed on the third model yielded a 20 variable model. These 20 variables are used for the fourth model and subsequent model testing as a comparison against models fit using all original variables. The most profitable model was the fourth model, yielding $11,649.50 through 1,302 mailings.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Coefficients:** | **Estimate** | **Significance** | | (Intercept) | 0.91854 | \*\*\* | | reg1 | 0.65734 | \*\*\* | | reg2 | 1.49913 | \*\*\* | | home | 1.39021 | \*\*\* | | chld | -2.40428 | \*\*\* | | I(hinc^2) | -1.09579 | \*\*\* | | genf | -0.08635 |  | | wrat | 0.49074 | \*\*\* | | I(wrat^2) | -0.41909 | \*\*\* | | incm | 0.54076 | \*\*\* | | inca | 0.11863 |  | | I(inca^2) | 0.00213 |  | | plow | -0.06942 |  | | tgif | 0.64033 | \*\*\* | | I(tgif^2) | -0.07046 | . | | lgif | -0.23234 | \* | | I(rgif^2) | 0.03872 |  | | tdon | -0.30501 | \*\*\* | | tlag | -0.54866 | \*\*\* | | agif | 0.16075 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Model** | **Validation Class** | | | | |  | **0** | **1** | **N. Mailings** | | **Model 1: All independent variables** | **0** | 600 | 21 | 1397 | | **1** | 419 | 978 | | **Model 2: All significant independent variables** | **0** | 418 | 17 | 1583 | | **1** | 601 | 982 | | **Model 3: All independent variables and additional terms** | **0** | 675 | 13 | 1330 | | **1** | 344 | 986 | | **Model 4: Backward subset selection** | **0** | 670 | 14 |  | |  | **1** | 349 | 985 | 1302 | |

# Model Technique 2: Logistic General Additive Model

The General Additive Model (GAM) approach to modeling creates a function for each variable, allowing non-linearity in the model. This approach can be used for classification problems easily with a logistic model. The first Logistic GAM model is fitted using all independent variables, and the second model fitted uses the 20 variables from the logistic model backward subset selection. The first model yields a profit of $11,381.00 with 1,400 mailings and the second model outperforms the first, yielding a profit of $11,649.50 with 1,302 mailings. The second model coefficients are identical to the logistic regression model from the previous section.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Validation Class** | | | |
|  | **0** | **1** | **Number of Mailings** |
| **Model 1: All independent variables** | **0** | 597 | 21 | 1400 |
| **1** | 422 | 978 |
| **Model 2: 20 Backward subset selection variables** | **0** | 700 | 16 | 1302 |
| **1** | 319 | 983 |

# Model Technique 3: Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a classification method similar to logistic regression that utilizes Bayes’ theorem as a classifier, and while it is theoretically more complex than the logistic model, in application it is similar in interpretability. The first LDA model is fit using all independent variables, yielding a profit of $11,643.50 through 1,363 mailings. A second LDA model fit using the 20 variables from the logistic model backward subset selection yields slightly lower profit of $11,639.50 through 1,336 mailings. Trade-offs exists between the two LDA models constructed. While the first model is more profitable, fewer individuals are correctly classified. The second model is less profitable, but fewer mailings are sent, and a greater number of individuals are correctly identified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Validation Class** | | | |
|  | **0** | **1** | **Number of Mailings** |
| **Model 1: All independent variables** | **0** | 647 | 8 | 1363 |
| **1** | 372 | 991 |
| **Model 2: 20 Backward subset selection variables** | **0** | 670 | 12 | 1336 |
| **1** | 349 | 987 |

# Model Technique 4: Quadratic Discriminant Analysis

Quadratic Discriminant Analysis (QDA) is a classification model almost identical to LDA, except that the covariance matrix is assumed to be specific to each class. This model can be a more flexible fit and tends to perform well on large datasets. The first QDA model is fit using all independent variables, followed by a second model utilizing the 20 backward subset selection variables used in prior sections of this study. Both models perform significantly worse than previous models. The second model required more mailings than the first with 1,421 and was least profitable at $11,136.00, compared to the first model’s profit of $11,229.50 and 1,396 mailings.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Validation Class** | | | |
|  | **0** | **1** | **Number of Mailings** |
| **Model 1: All independent variables** | **0** | 590 | 32 | 1396 |
| **1** | 429 | 967 |
| **Model 2: 20 Backward subset selection variables** | **0** | 560 | 37 | 1421 |
| **1** | 459 | 962 |

# Model Technique 5: K-Nearest Neighbor Classification

Although not necessarily the most interpretable of algorithms, the K-Nearest Neighbor classification method (referred to as KNN) is strong as it is one of the few completely non-parametric methods available for classification. Since there is little information known about the exact decision boundary for whether or not an individual is a donor or not, this technique is certainly on that should be tested. For this process, three models are fit utilizing different values of K: 1, 10 and 100. Although the classifier with K = 100 performs the best, it is ultimately somewhat expected since it will more likely have much higher bias towards the training data set. Despite this, the performance on the validation data still does better than the other two classifiers, maximizing profit at $11,299.50 through 1,390 mailings.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Validation Class** | | | |
|  | **0** | **1** | **Number of Mailings** |
| **K = 1** | **0** | 738 | 164 | 1116 |
| **1** | 281 | 835 |
| **K = 10** | **0** | 709 | 59 | 1250 |
| **1** | 310 | 940 |
| **k = 100** | **0** | 600 | 28 | 1390 |
| **1** | 419 | 971 |

In **[Table X]** above, though, one can observe that although the number of correctly classified individuals goes up, so does the number of incorrectly classified individuals. This larger profit ultimately results in a lot more wasted mailings, as well.

# Model Technique 6: Decision Tree

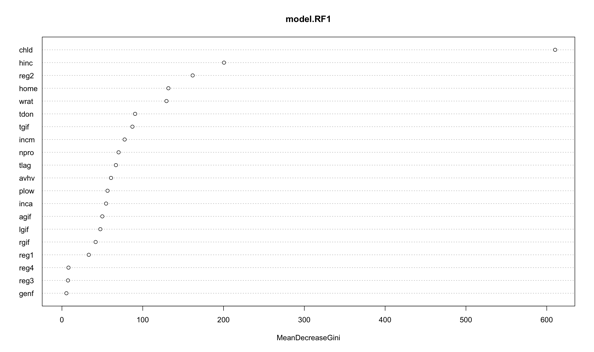
Tree-based methods are considered simple and useful for interpretation, thus a simple decision tree is considered for classifying whether or not an individual is a donor. Since the process of building a decision tree includes a technique for pruning the resulting tree, all 20 variables are passed in for model fitting. Although the initial decision tree fit results in 1362 mailings and a profit of $11,413.50, it is good practice to examine pruning the tree to ensure the original tree is not limited in performance due to over-fitting by the training data.

|  |  |
| --- | --- |
|  |  |

Although **[Figure X]** demonstrates that the best decision tree is of size 10, the resulting tree from fitting that model is the exact same as the original decision tree model fit, capping this technique off at a $11,413.50 profit.

# Model Technique 7: Random Forest

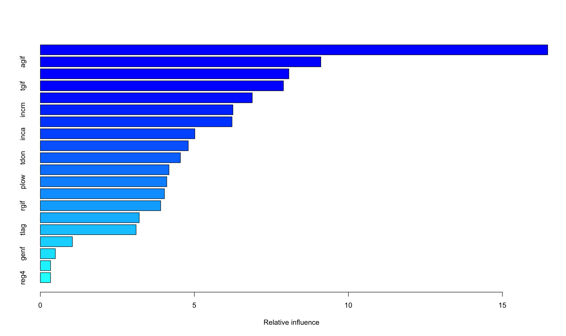
Having observed that the single, pruned decision tree still hasn’t maximized profit, a logical next step would be to try a method for aggregating several decision trees to enhance the performance of a single tree. Although there are three common techniques used to achieve this (bagging, boosting and random forests), the first of two techniques to be tested for this analysis is the Random Forest modeling technique. The model fit results in only 1,055 total mailings, maxing out its profit at $11,099.50. Although this technique is demonstrating that it isn’t the best in terms of profit, the illustrating in **[Figure X]** below is incredibly useful in that it provide a clear visualization of the variables that are contributing the most to whether or not an individual will donate or not.



# Model Technique 8: Boosted Tree

With the resultant random forest performing so poorly in contrast to the individual decision tree, one additional attempt at a multiple tree aggregation method will be performed. This method, referred to as boosting, learns more slowly and tends to reduce the likelihood of over-fitting as well. The boosted tree results in a total of 1308 mailings, soaring the resulting profit to $11,565.00. Of all tree based classification methods built, this is by far the highest performer. In **[Figure X]** below, one can observe a similar visualization of the variables based on their relative influence on the classification of whether or not an individual is a donor or not. Similarly to the other two tree based models, the most important variable is how many children the potential donor has.

|  |  |
| --- | --- |
| Var | Relative Influence |
| chld | 16.4728955 |
| agif | 9.1065005 |
| avhv | 8.0688407 |
| tgif | 7.893357 |
| hinc | 6.8809433 |
| incm | 6.2528839 |
| npro | 6.224251 |
| inca | 5.0164309 |
| wrat | 4.8032304 |
| tdon | 4.5479742 |
| lgif | 4.1767537 |
| plow | 4.1053603 |
| reg2 | 4.0291227 |



# Model Technique 9: Support Vector Machine

The last but certainly not least classifier to be tested on this data is the Support Vector Machine, first: without any tuning utilizing a linear kernel and second: utilizing cross validation to determine optimal parameters as well as utilizing a more likely applicable radial kernel. As expected, the first model tanks in comparison to all previously fit classification models with a maximal profit of $10,534, mostly due to suggesting far too many mailings at 1,925. The tuned model, utilizing the radial kernel performs substantially better, raking in a potential profit of $11,336.50 on the validation data through only 1,444 mailings.

## Regression Models and Analysis

# Model Technique 1: Least Squares Regression

We fit several least squares regression models by finding the models with the best Adjusted R-Squared, Mallows’ Cp, or BIC values. The model with the best Adjusted R-Squared value contained 15 predictors; the model with the best Mallows’ Cp contained 14 predictors; and the model with the best BIC contained 11 predictors. **Table X** below is a comparison of how each of these least squares regression models performed on the validation data.

|  |  |  |
| --- | --- | --- |
| Model | Validation MPE | Standard Error of Validation MPE |
| Adjusted R-Squared | 1.508110 | 0.1570488 |
| Mallows' Cp | 1.508530 | 0.1568465 |
| BIC | 1.527695 | 0.1596775 |

In terms of Validation MPE, the Adjusted R-Squared model outperformed the other two models. **Table X** shows the coefficient estimates for the Adjusted R-Squared model based on the training data. We can see that all of the predictors except reg1 are statistically significant.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable | Estimate | Std. Error | t value | Pr(>|t|) | Signif. |
| (Intercept) | 14.18794 | 0.04091 | 346.79 | < 2e-16 | \*\*\* |
| reg1 | -0.03695 | 0.03616 | -1.022 | 0.307 |  |
| reg2 | -0.08396 | 0.03922 | -2.141 | 0.0324 | \* |
| reg3 | 0.31674 | 0.03699 | 8.563 | < 2e-16 | \*\*\* |
| reg4 | 0.64296 | 0.03805 | 16.896 | < 2e-16 | \*\*\* |
| home | 0.22464 | 0.05561 | 4.04 | 0.0000555 | \*\*\* |
| chld | -0.59255 | 0.03437 | -17.239 | < 2e-16 | \*\*\* |
| hinc | 0.51018 | 0.03645 | 13.996 | < 2e-16 | \*\*\* |
| genf | -0.05985 | 0.02613 | -2.291 | 0.0221 | \* |
| incm | 0.43668 | 0.04431 | 9.854 | < 2e-16 | \*\*\* |
| plow | 0.39436 | 0.04793 | 8.228 | 3.42E-16 | \*\*\* |
| tgif | 0.20083 | 0.02674 | 7.511 | 8.85E-14 | \*\*\* |
| lgif | 0.3893 | 0.05857 | 6.647 | 3.86E-11 | \*\*\* |
| rgif | 0.47417 | 0.05093 | 9.31 | < 2e-16 | \*\*\* |
| tdon | 0.07134 | 0.03205 | 2.226 | 0.0261 | \* |
| agif | 0.38639 | 0.04847 | 7.971 | 2.63E-15 | \*\*\* |
| Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1 | | | | | |

The coefficient estimates in **Table X** reveals that all but reg1, genf, and tdon are statistically significant for the Adjusted R-Squared model using the validation data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable | Estimate | Std. Error | t value | Pr(>|t|) | Signif. |
| (Intercept) | 14.1476 | 0.06392 | 221.326 | < 2e-16 | \*\*\* |
| reg1 | -0.02995 | 0.0573 | -0.523 | 0.601237 |  |
| reg2 | -0.16548 | 0.06066 | -2.728 | 0.006488 | \*\* |
| reg3 | 0.26577 | 0.05843 | 4.549 | 0.00000607 | \*\*\* |
| reg4 | 0.75513 | 0.05814 | 12.989 | < 2e-16 | \*\*\* |
| home | 0.32659 | 0.09556 | 3.418 | 0.000657 | \*\*\* |
| chld | -0.53807 | 0.05235 | -10.279 | < 2e-16 | \*\*\* |
| hinc | 0.51402 | 0.05472 | 9.393 | < 2e-16 | \*\*\* |
| genf | 0.04099 | 0.03973 | 1.032 | 0.302414 |  |
| incm | 0.47313 | 0.06933 | 6.824 | 1.54E-11 | \*\*\* |
| plow | 0.48097 | 0.07302 | 6.587 | 7.32E-11 | \*\*\* |
| tgif | 0.2179 | 0.04079 | 5.342 | 0.000000114 | \*\*\* |
| lgif | 0.53547 | 0.08875 | 6.033 | 2.27E-09 | \*\*\* |
| rgif | 0.33297 | 0.0767 | 4.341 | 0.0000156 | \*\*\* |
| tdon | 0.03494 | 0.05083 | 0.687 | 0.491963 |  |
| agif | 0.40689 | 0.07144 | 5.696 | 1.62E-08 | \*\*\* |
| Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1 | | | | | |

# Model Technique 2: Best Subset Regression with K-Fold Cross Validation

For this model, we used 20-fold cross-validation to find the best subset selection model with the lowest mean cross-validation error. **Figure X** shows that the model with 18 variables has the lowest mean cross-validation error.

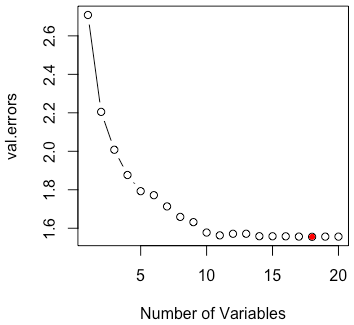


Figure X: Best Subset Selection

Using Cross-Validation

**Table X** shows the model coefficients using the training data. Similar to the Adjusted R-Squared regression model, reg1 is not statistically significant. However, all the other variables except wrat, inca, and tlag are significant.

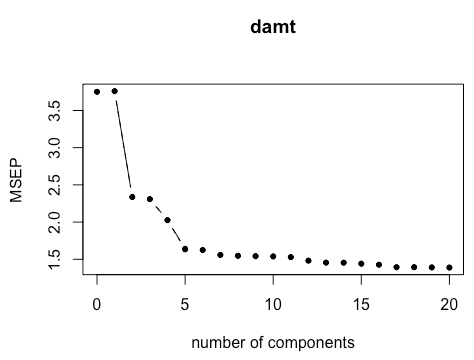
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable | Estimate | Std. Error | t value | Pr(>|t|) | Signif. |
| (Intercept) | 14.18688 | 0.0436 | 325.424 | < 2e-16 | \*\*\* |
| reg1 | -0.03719 | 0.0362 | -1.027 | 0.3044 |  |
| reg2 | -0.08488 | 0.03932 | -2.159 | 0.031 | \* |
| reg3 | 0.31578 | 0.03701 | 8.531 | < 2e-16 | \*\*\* |
| reg4 | 0.64187 | 0.03808 | 16.854 | < 2e-16 | \*\*\* |
| home | 0.22316 | 0.05568 | 4.008 | 0.0000636 | \*\*\* |
| chld | -0.59284 | 0.03487 | -17.001 | < 2e-16 | \*\*\* |
| hinc | 0.51069 | 0.03654 | 13.975 | < 2e-16 | \*\*\* |
| wrat | 0.02196 | 0.03807 | 0.577 | 0.5641 |  |
| genf | -0.05986 | 0.02614 | -2.29 | 0.0222 | \* |
| incm | 0.41271 | 0.05625 | 7.338 | 3.16E-13 | \*\*\* |
| inca | 0.03616 | 0.04951 | 0.73 | 0.4652 |  |
| plow | 0.40391 | 0.04952 | 8.157 | 6.03E-16 | \*\*\* |
| tgif | 0.20043 | 0.02677 | 7.488 | 1.05E-13 | \*\*\* |
| lgif | 0.38749 | 0.05862 | 6.611 | 4.91E-11 | \*\*\* |
| rgif | 0.47543 | 0.05096 | 9.329 | < 2e-16 | \*\*\* |
| tdon | 0.07232 | 0.03208 | 2.255 | 0.0243 | \* |
| tlag | 0.02516 | 0.0309 | 0.814 | 0.4157 |  |
| agif | 0.38728 | 0.0485 | 7.985 | 2.36E-15 | \*\*\* |
| Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1 | | | | | |

**Table X** shows the model coefficients using the validation data. The validation data shows a slight change in significant variables. Whereas genf and tdon were significant to an alpha of 0.05 in the training data, they are not significant in the validation data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable | Estimate | Std. Error | t value | Pr(>|t|) | Signif. |
| (Intercept) | 14.13508 | 0.06693 | 211.187 | < 2e-16 | \*\*\* |
| reg1 | -0.03459 | 0.05746 | -0.602 | 0.547343 |  |
| reg2 | -0.16425 | 0.06078 | -2.703 | 0.007 | \*\* |
| reg3 | 0.2614 | 0.05855 | 4.465 | 0.00000894 | \*\*\* |
| reg4 | 0.75418 | 0.05826 | 12.945 | < 2e-16 | \*\*\* |
| home | 0.33059 | 0.09579 | 3.451 | 0.000582 | \*\*\* |
| chld | -0.54767 | 0.05354 | -10.229 | < 2e-16 | \*\*\* |
| hinc | 0.51401 | 0.05477 | 9.385 | < 2e-16 | \*\*\* |
| wrat | 0.04209 | 0.05284 | 0.797 | 0.425852 |  |
| genf | 0.03928 | 0.04008 | 0.98 | 0.327336 |  |
| incm | 0.41257 | 0.08568 | 4.815 | 0.0000017 | \*\*\* |
| inca | 0.09238 | 0.0771 | 1.198 | 0.231127 |  |
| plow | 0.50343 | 0.07607 | 6.618 | 6E-11 | \*\*\* |
| tgif | 0.21986 | 0.04088 | 5.378 | 0.000000094 | \*\*\* |
| lgif | 0.53057 | 0.08887 | 5.97 | 3.31E-09 | \*\*\* |
| rgif | 0.33046 | 0.07692 | 4.296 | 0.0000191 | \*\*\* |
| tdon | 0.03997 | 0.05099 | 0.784 | 0.433341 |  |
| tlag | 0.01897 | 0.043 | 0.441 | 0.659236 |  |
| agif | 0.41539 | 0.0717 | 5.794 | 9.27E-09 | \*\*\* |
| Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1 | | | | | |

# Model Technique 3: Principal Components

In order to choose the number of principal components, we looked at **Figure X**. The lowest MSEP value is around 20 components, but the “elbow” of the plot appears around 5 components. We decided to fit models using 5, 15, and 20 components to determine the optimal number of components.

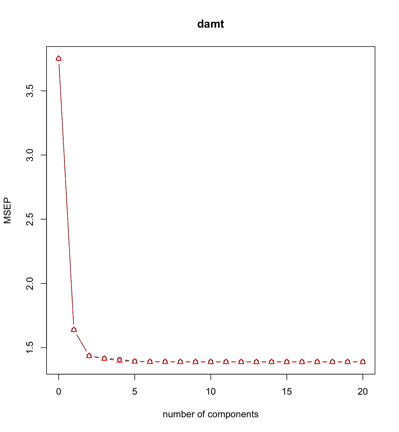


In **Table X**, we can see that the model using 20 principal components performed best as it had the smallest MPE. However, we could choose the model containing 15 components if simplicity was a priority.

|  |  |  |
| --- | --- | --- |
| Number of Components | Validation MPE | Standard Error of Validation MPE |
| 5 | 1.812705 | 0.1688532 |
| 15 | 1.598671 | 0.1607489 |
| 20 | 1.556378 | 0.1612150 |

# Model Technique 4: Partial Least Squares

The fourth model in the regression section is a partial least squares model. Similarly to Model technique 3 in this section, Model Technique 4 requires us to select the number of components that we think are appropriate based on the following plot:

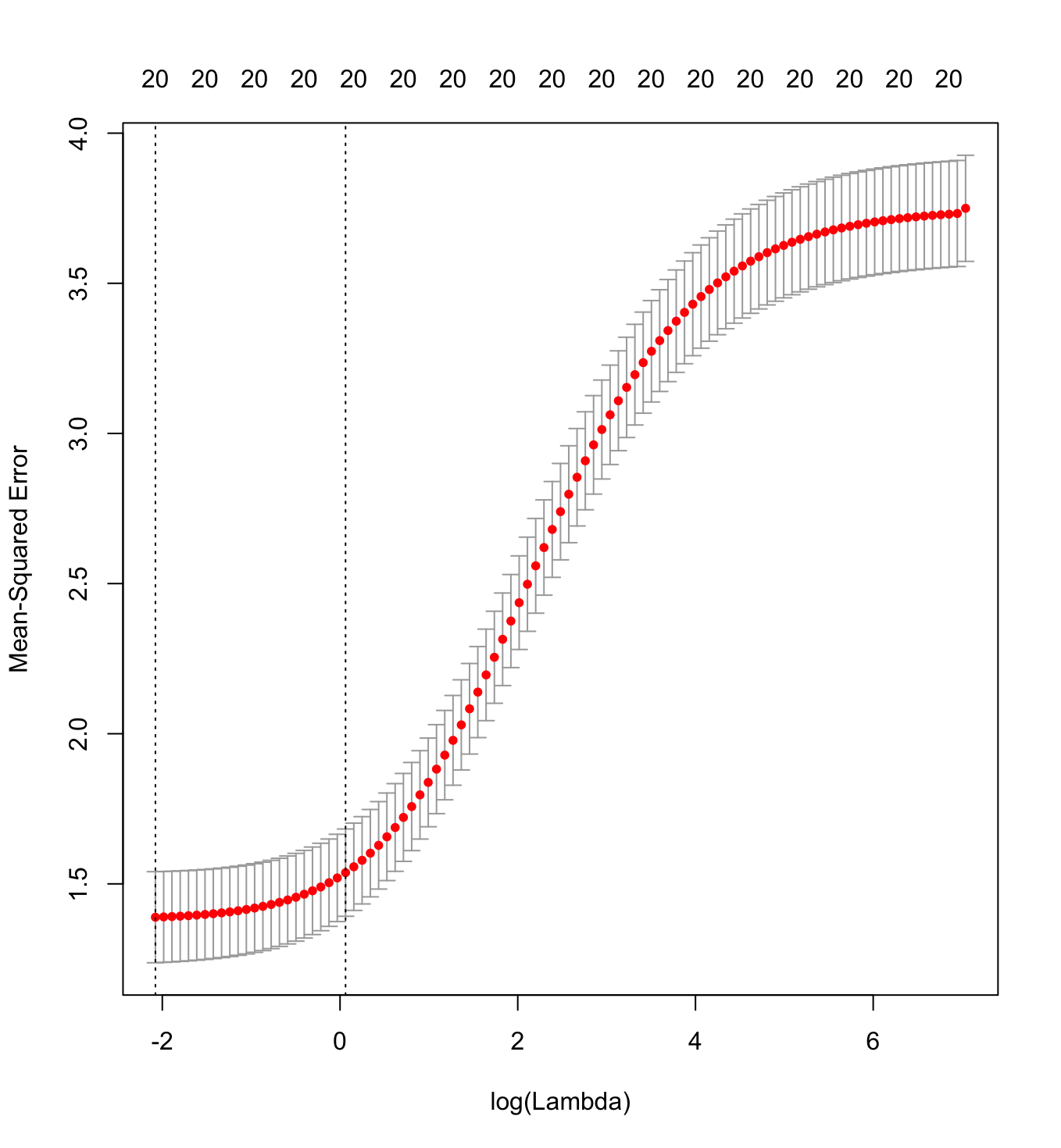


Looking at the plot for partial least squares, we can see that there is a large drop down to 3 components, but not much movement after that. This leads us to believe that 3 components are enough. The results for this model are below:

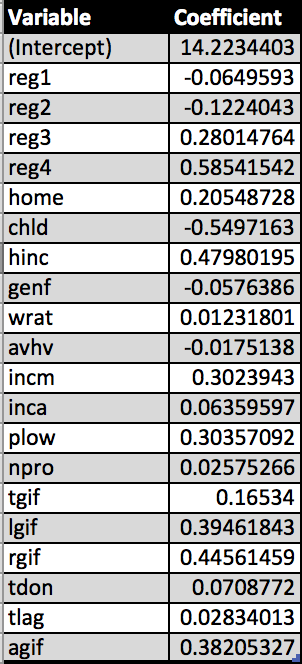
|  |  |  |
| --- | --- | --- |
| # of Components | MPE | Standard Error |
| 3 | 1.592151 | 0.1613484 |

# Model Technique 5: Ridge Regression

The fifth model technique in the regression section is ridge regression. For this model type, we first need to select the best lambda. In **[Figure X]**, one can see the MSE is smallest when log(Lambda) is smallest. The first vertical line from the left of the plot signifies the smallest MSE, while the second vertical line represents one standard deviation from the minimum MSE. The number 20 at the top of the plot indicates that all twenty predictor variables are present in the model regardless of the lambda value.



Using the best lambda equal to 0.1252296 to estimate the coefficients in **[Table X]** below.

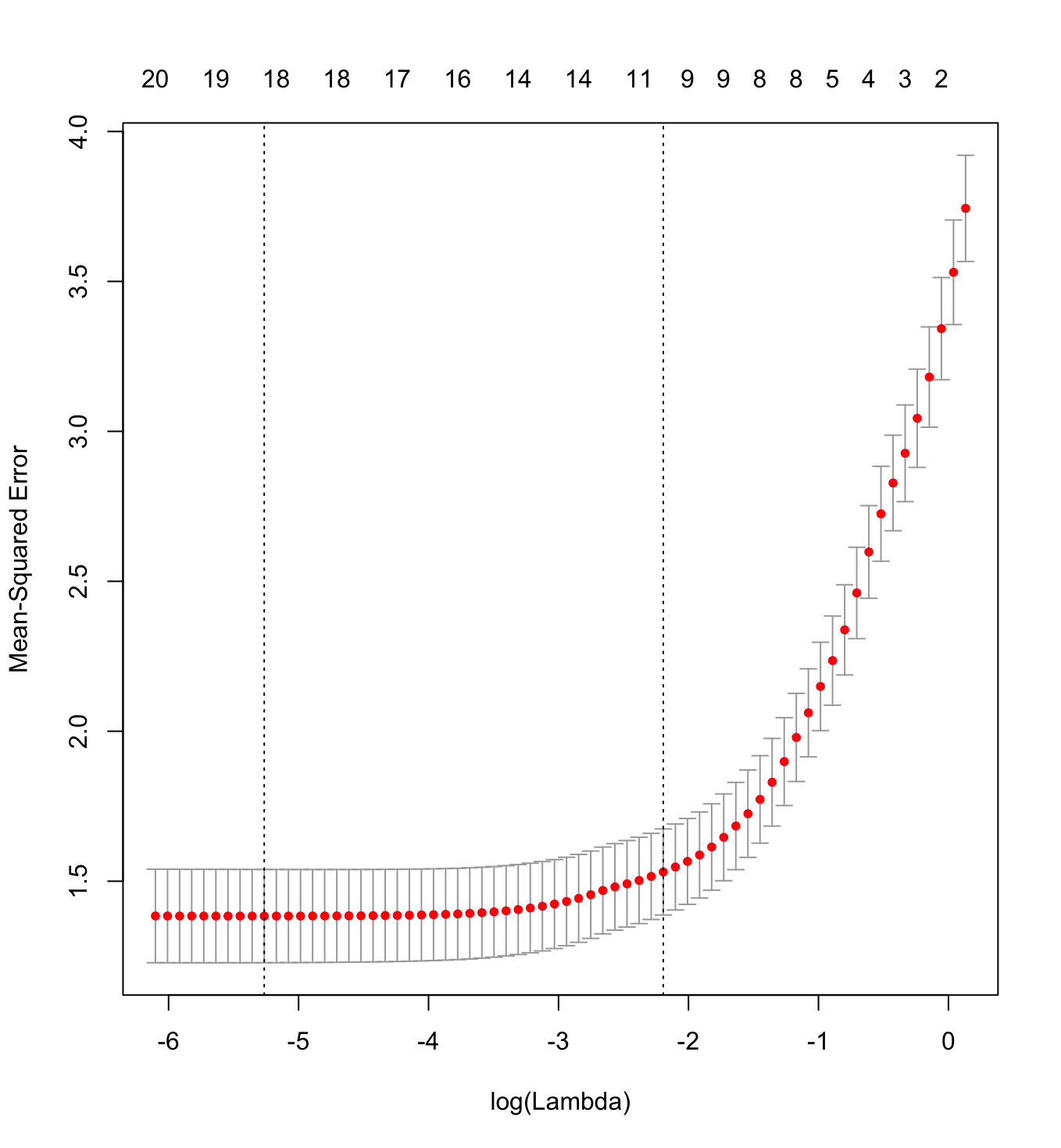


This model is used to make the predictions and compute the following errors:

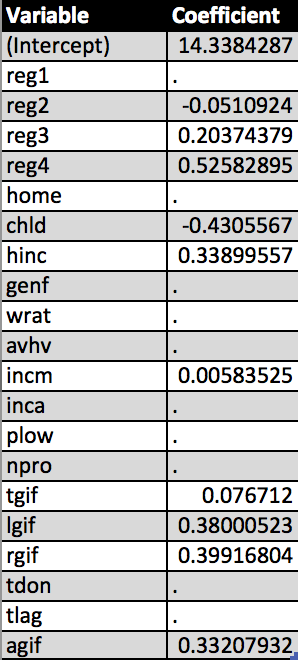
|  |  |
| --- | --- |
| MPE | Standard Error |
| 1.572113 | 0.1627705 |

# Model Technique 6: Lasso

The last model technique in this section is lasso. Looking at the second vertical line from the left in **[Figure X]**, one can see that the number of predictors in the model is 10, as denoted by the number at the top of the plot.



Again we first need to select the best lambda using cross-validation. In this case the best lambda is 0.005174504, which produces the following modeling coefficients.



Using this lambda, we make the predictions and compute the following errors:

|  |  |
| --- | --- |
| MPE | Standard Error |
| 1.562046 | 0.1611463 |

## Results

# Classification Models

As discussed, there have been a total of nine classification techniques tested on this data to determine which model is the best for classifying individuals as donors. Utilizing the number of mailings and their estimated profit on the validation data set as decision metrics, the best model would be the one that maximizes profits through minimizing extraneous mailings. Although a little surprising, the model with the best performance on the validation data is the Logistic Model that utilizes a subset of predictors based on a backwards selection variable technique. The Generalized Additive Logistic Regression Model results in the same model fit via the original logistic model with backward selection. The logistic regression model boasts the additional benefit of being highly interpretable as well. Therefore, this model will used to classify individuals on the test set on whether or not they are donors.

|  |  |  |
| --- | --- | --- |
| **# Mailings** | **Profit** | **Modeling Type** |
| 1397 | $11,387.00 | Log1 |
| 1583 | $11,073.00 | Log 1a |
| 1302 | $11,649.50 | Log 1b |
| 1396 | $11,389.00 | Log GAM1 |
| 1302 | $11,649.50 | Log GAM1a |
| 1363 | $11,642.50 | LDA1 |
| 1336 | $11,639.50 | LDA1a |
| 1396 | $11,229.50 | QDA |
| 1421 | $11,107.00 | QDA1a |
| 1116 | $9,875.50 | KNN |
| 1250 | $11,130.00 | KNN1a |
| 1390 | $11,299.50 | KNN1b |
| 1362 | $11,413.50 | Unaltered Tree & Pruned Tree |
| 1487 | $11,381.00 | Tree with subset vars |
| 1308 | $11,565.00 | Boosted Tree |
| 1055 | $11,099.50 | RF |
| 1925 | $10,534.00 | Linear SVM (untuned) |
| 1366 | $11,536.00 | Radial SVM (tuned) |

# Regression Models

|  |  |  |
| --- | --- | --- |
| Model | Test MSE | Standard Error of Test MSE |
| 1 | 3111.265 | 361.0908 |
| 2 | 3095.483 | 369.7526 |
| 3 | 3095.483 | 369.7526 |
| 4 | 3070.870 | 350.5467 |
| 5 | 2920.041 | 346.2248 |
| 6 |  |  |

Model 1 (least squares regression containing all ten predictors) resulted in the worst test MSE, although its test MSE standard error was slightly better than the test MSE standard error for Models 2 and 3. As mentioned earlier, Model 2 (best subset selection using BIC) and Model 3 (best subset selection using 10-fold cross-validation) resulted in the same six-variable model with identical coefficient estimates and test errors. Model 4 (ridge regression using 10-fold cross-validation) performed slightly better than Models 1, 2, and 3. It, however, is a more complex model than Models 2 and 3 because it uses all ten predictors rather than just six. Model 5 (lasso using 10-fold cross-validation) resulted in the smallest test MSE and test MSE standard error of all the models. It contains only six predictors. Because of its small test errors and relative simplicity, Model 5 can be considered the best model.

## Conclusion

After fitting various machine learning modeling techniques, including least squares regression, best subset selection using BIC, best subset selection using 10-fold cross validation, ridge regression using 10-fold cross validation, and lasso using 10-fold cross validation, I discovered that the lasso model using 10-fold cross validation performed best in predicting the progression of diabetes one year after baseline. This model contained the predictor variables sex, bmi, map, hdl, ltg, and glu. Before deploying the model, I recommend further model testing and consulting a diabetes expert to determine if the predictors included in the model make medical sense. If the model performs well in additional tests and receives the approval of a medical expert, it should be safe to deploy the model on new data.

## Appendix A: R Code

1. Please note: we kept the transformed variable names the same as the original variable names for our models, but we added a “.t” to the end of each variable name in Figure 1 to clearly distinguish between the original and transformed variables. [↑](#footnote-ref-1)