Predicting Donors and Donation Amounts

ADTA 5230 Data Analytics II Section 410

Group 6

Sana Ambreen, Ashley Chesser, Yitian Liu, & Kiera Wingo

## Introduction

Our team has been hired by a nonprofit organization in need of cost-savings solutions for their direct marketing campaigns soliciting donations. Their response rate is currently around 10%, with the average donation being $14.50. However, each mailer, which includes a personalized gift, costs the nonprofit $2.00. With all costs factored in, sending a piece of mail to everyone is actually losing them about $0.55 per mailer. This is why the nonprofit has reached out for help in developing predictive models to improve how they run their mailing campaigns.

The first step will be building three classification models that predict whether someone is likely to donate or not. The goal here is to maximize the number of donors who receive the mailer, while also minimizing the mailer recipients who do not end up donating.

Next, we will have three regression models that estimate how much someone might give if they do donate. These models will help the organization avoid spending money on people who are unlikely to give, while spending on those who are more likely to contribute. We will be working with a pre-processed dataset where donors and non-donors are equally represented. All the data preparation, modeling, and evaluation will be done using SAS Enterprise Miner.

By the end of this project, we hope to provide the nonprofit with useful data they can use to make more informed mailing decisions for campaigns. Even small improvements in targeting the right people could lead to significant profit increases and less wasted money and efforts.

## Exploratory Data Analysis

### Data Overview

The nonprofit data consists of 6002 rows and 20 columns, and contains two target variables: donr, a binary variable detailing whether an individual has donated or not, and damt, a numerical variable detailing the amount each donor has given. Figure 1 showcases the first five rows of data from the dataset and Table 1 gives a detailed account of each variable and its description.

```
data.head()
```

| | ID | region | ownd | kids | inc | sex | wlth | hv | incmed | incavg | low | npro | gifdol | gifl | gifr | mdon | lag | gifa | donr | damt |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | ter3 | 1 | 1 | 4 | 1 | 8 | 302 | 76 | 82 | 0 | 20 | 81 | 81 | 19 | 17 | 6 | 21.05 | 0 | 0 |
| **1** | 2 | ter3 | 1 | 2 | 4 | 0 | 8 | 262 | 130 | 130 | 1 | 95 | 156 | 16 | 17 | 19 | 3 | 13.26 | 1 | 15 |
| **2** | 5 | ter3 | 1 | 0 | 4 | 1 | 4 | 295 | 39 | 71 | 14 | 85 | 132 | 15 | 10 | 10 | 6 | 12.07 | 1 | 17 |
| **3** | 6 | ter2 | 1 | 1 | 5 | 0 | 9 | 114 | 17 | 25 | 44 | 83 | 131 | 5 | 3 | 13 | 4 | 4.12 | 1 | 12 |
| **4** | 7 | ter5 | 1 | 3 | 4 | 0 | 8 | 145 | 39 | 42 | 10 | 50 | 74 | 6 | 5 | 22 | 3 | 6.50 | 0 | 0 |

*Figure 1*

**Table 1**

| Variable Name | Data Type | Description |
|---|---|---|
| id number | Integer | Do NOT use this as a predictor variable in any models |
| region | Object | five geographic regions including ter1, ter2, ter3, ter4, ter5 |
| ownd | Binary | (1 = homeowner, 0 = not a homeowner) |
| kids | Integer | Number of children |
| inc | Object | Household income (7 categories) |
| sex | Object | Gender (0 = Male, 1 = Female) |
| wlth | Integer | Wealth Rating (Wealth rating uses median family income and population statistics from each area to index relative wealth within each state. The segments are denoted 0-9, with 9 being the highest wealth group and 0 being the lowest.) |
| hv | Numeric | Average Home Value in potential donor's neighborhood in $ thousands |
| incmed | Numeric | Median Family Income in potential donor's neighborhood in $ thousands |
| incavg | Numeric | Average Family Income in potential donor's neighborhood in $ thousands |
| low | Numeric | Percent categorized as “low income” in potential donor's neighborhood |
| npro | Integer | Lifetime number of promotions received to date |
| gifdol | Numeric | Dollar amount of lifetime gifts to date |
| gifl | Numeric | Dollar amount of largest gift to date |
| gifr | Numeric | Dollar amount of most recent gift |
| mdon | Integer | Number of months since last donation |
| lag | Integer | Number of months between first and second gift |
| gifa | Numeric | Average dollar amount of gifts to date |
| donr | Binary | Classification Response Variable (1 = Donor, 0 = Non-donor) |

| | | |
|---|---|---|
| damt | Numeric | Prediction Response Variable (Donation Amount in $). |

## Descriptive Statistics

Figure 2 showcases the descriptive statistics of each numerical variable in the data. From the 6002 total observations, incmed, low, gifdol, and gifl have high averages while hv and gifdol show the highest maximum values.

| | Count | Mean | Median | Std | Min | Max |
|---|---|---|---|---|---|---|
| ID | 6002 | 3978.91 | 3945.5 | 2301.81 | 1.00 | 8009.00 |
| ownd | 6002 | 0.88 | 1.0 | 0.32 | 0.00 | 1.00 |
| kids | 6002 | 1.58 | 2.0 | 1.41 | 0.00 | 5.00 |
| inc | 6002 | 3.94 | 4.0 | 1.40 | 1.00 | 7.00 |
| sex | 6002 | 0.61 | 1.0 | 0.49 | 0.00 | 1.00 |
| wlth | 6002 | 7.02 | 8.0 | 2.33 | 0.00 | 9.00 |
| hv | 6002 | 183.91 | 170.0 | 72.77 | 51.00 | 710.00 |
| incmed | 6002 | 43.95 | 38.0 | 24.66 | 3.00 | 287.00 |
| incavg | 6002 | 56.79 | 52.0 | 24.83 | 14.00 | 287.00 |
| low | 6002 | 13.89 | 10.0 | 13.10 | 0.00 | 87.00 |
| npro | 6002 | 61.35 | 59.0 | 30.31 | 2.00 | 164.00 |
| gifdol | 6002 | 115.80 | 91.0 | 86.54 | 23.00 | 1974.00 |
| gifl | 6002 | 22.98 | 16.0 | 29.40 | 3.00 | 642.00 |
| gifr | 6002 | 15.65 | 12.0 | 12.42 | 1.00 | 173.00 |
| mdon | 6002 | 18.79 | 18.0 | 5.60 | 5.00 | 40.00 |
| lag | 6002 | 6.32 | 5.0 | 3.64 | 1.00 | 34.00 |
| gifa | 6002 | 11.68 | 10.2 | 6.53 | 1.89 | 72.27 |
| donr | 6002 | 0.50 | 0.0 | 0.50 | 0.00 | 1.00 |
| damt | 6002 | 7.21 | 0.0 | 7.36 | 0.00 | 27.00 |

*Figure 2*

## Data Quality Inspection & Univariate Analysis

### *Missing Values*

As Figure 3 details, there are no missing values in the dataset and no action was taken to impute any data.

```
data.isnull().sum()
```

```
ID        0
region    0
ownd      0
kids      0
inc       0
sex       0
wlth      0
hv        0
incmed    0
incavg    0
low       0
npro      0
gifdol    0
gifl      0
gifr      0
mdon      0
lag       0
gifa      0
donr      0
damt      0
dtype: int64
```

*Figure 3*

## Univariate Analysis

According to Table 1, there are several core variables that need to be analyzed. In the categorical variables, these variables need to be analyzed first: inc, wlth, donr.

***Variable: donr***

As we can see from Figure 4, the proportion of donation and non-donation population are very similar. Donation population is 49.9% and non-donation population is 50.1%, which indicates that, in the market, these two types of people are comparable.

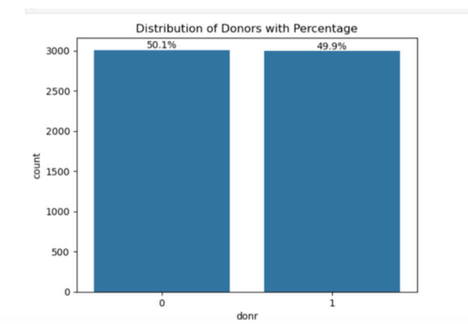



*Figure 4*

***Variable: inc***

As we can see from Figure 5, the household income level 4 is in the dominant position at 45.9%. The second and third are level 5 at 14.8% and level 3 at 10.1%, respectively. This is interesting that the sample dataset is concentrated in the middle level of household income.

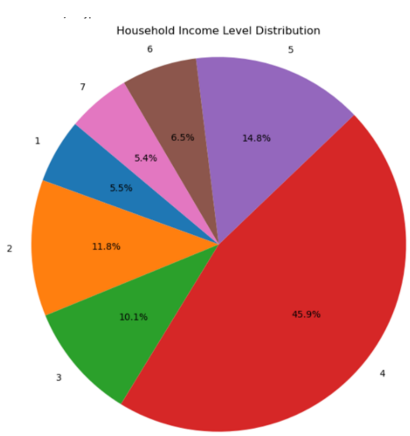



*Figure 5*

***Variable: wlth***

According to Figure 6, the dominant wealth rate is level 8 at 38.6%, and the second and third are level 9 at 27.1% and level 6 at 6.9%, respectively. Besides, levels 0, 1, 2, and 3 all account for a very small proportion of the wealth rate. The high level of wealth population is much more than the low level of wealth, which should influence the model training.

*Figure 6*

***Outlier Detection***

Outliers sometimes will heavily impact on the accuracy of data analysis. There are 2 core data that we need to check because they are prone to have outliers which may influence business understanding. After the inspection (Figure 7), for hv (average home value), the majority part of hv concentrates in 150,000-200,000, but some outliers at the right tail are over 700,000. These outliers are reasonable because it belongs to the high-end market Luxury housing range.

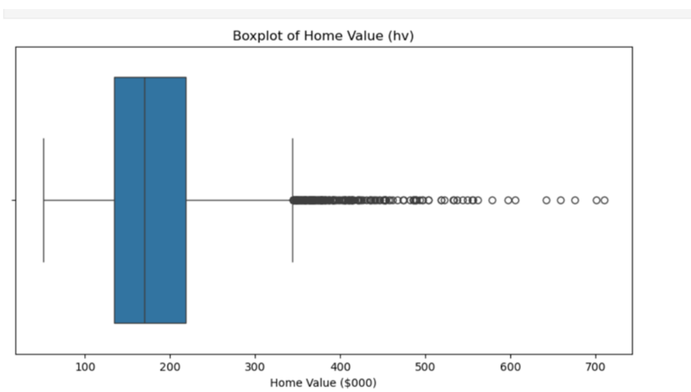



*Figure 7*

For the gifdol (Lifetime Gift Amount, Figure 8), the majority is concentrated below 250, but some outliers at the right tail are over 2,000. Both of these outliers are needed to care about the overfitting issue.

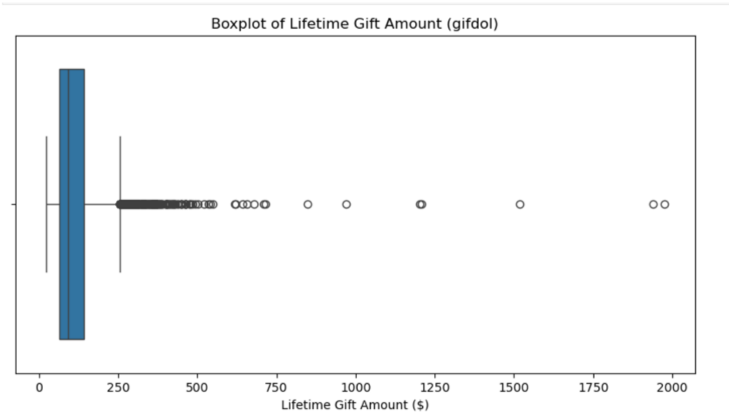



*Figure 8*

**Two Variable Analysis**

There are three pairs of variables that necessitated further analysis to determine preliminary relationships between them. The first was Hv versus gifdol, which showed that the population of lower- and middle-class homes are most prone to donation. Secondly was Wlth versus donr, which revealed that the proportion of donations among high wealth people is higher than non-donation, meaning the wealthy population is more prone to donate. Finally, looking at region versus damt, Figure 9 determined that the distribution of donations in regions discovered Ter4 has the highest donations and Ter5 has the lowest one. Region also proved to be an influencing factor on donation amount.

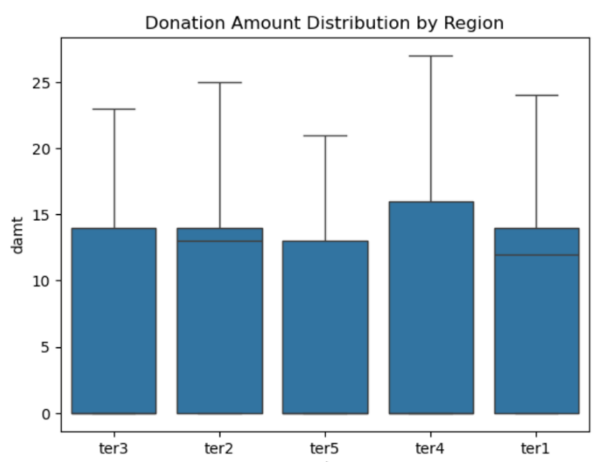



*Figure 9*

## Data Preparation

After initial exploration into the dataset, it was determined that no missing values were present. While there were a few outliers for the variables hv and gifdol, analysis revealed that they are not significant enough to disrupt calculations. As we can see from Figure 10, the formation of each column is accurate and does not need to be modified.

```
[14]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6002 entries, 0 to 6001
Data columns (total 20 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   ID      6002 non-null   int64
 1   region  6002 non-null   object
 2   ownd    6002 non-null   int64
 3   kids    6002 non-null   int64
 4   inc     6002 non-null   int64
 5   sex     6002 non-null   int64
 6   wlth    6002 non-null   int64
 7   hv      6002 non-null   int64
 8   incmed  6002 non-null   int64
 9   incavg  6002 non-null   int64
 10  low     6002 non-null   int64
 11  npro    6002 non-null   int64
 12  gifdol  6002 non-null   int64
 13  gifl    6002 non-null   int64
 14  gifr    6002 non-null   int64
 15  mdon    6002 non-null   int64
 16  lag     6002 non-null   int64
 17  gifa    6002 non-null   float64
 18  donr    6002 non-null   int64
 19  damt    6002 non-null   int64
dtypes: float64(1), int64(18), object(1)
memory usage: 937.9+ KB
```

*Figure 10*

## Modeling

### Classification

The Classification variable we want to predict here is donr (0 = non donor and 1 = donr). Various classification models can be applied to predict the outcome. Here, three classification models have been chosen for predicting the target variable. They are Logistic Regression, Neural Networks, and Random Forest.

#### *Classification Model Selection*

Chosen for being a generalized linear model, Logistic Regression was employed to predict the binary target variable donr. The benefits of utilizing the algorithm include its independent observations, exclusion of irrelevant predictors, and its assumption of linearity between the log-odds of the outcome and each continuous predictor (GeeksforGeeks, 2025). Logistic Regression’s ease of implementation, accuracy for simple data, and lack of required standardization were also factors in its selection for this project.

The second classification model, Neural Networks, was selected for its ability to model nonlinear data, speed of predictions, and ability to handle big data. The potential downsides of this model include the inability to understand the influence of each independent variable on the dependent variables, the lack of variable selection, and the time computational power required to build and train this model.

The final classification model selected was Random Forest. This nonparametric model was selected for its ability to model continuous and categorical data, lack of required standardization or normalization, and insensitivity to outliers. The potential downside to using this method is the complexity as trees increase in size and an extended training period.

***Classification Modelling Process***

After the data was cleaned and prepared, the dataset was imported into the File Node of SAS EM. The target variable (donr) was defined and the measurement levels of all the predictors were determined as shown in Figure 11. After that, the data partition node was added to divide the imported dataset into Train (70%) and Validation (30%).

Columns: Label    Mining

| Name | Role | Level | Report | Order | Drop | Lower Limit | Upper Limit |
|---|---|---|---|---|---|---|---|
| damt | Rejected | Interval | No | | No | . | . |
| donr | Target | Binary | No | | No | . | . |
| gifa | Input | Interval | No | | No | . | . |
| gifdol | Input | Interval | No | | No | . | . |
| gifl | Input | Interval | No | | No | . | . |
| gifr | Input | Interval | No | | No | . | . |
| hv | Input | Interval | No | | No | . | . |
| ID | Rejected | Nominal | No | | No | . | . |
| inc | Input | Ordinal | No | | No | . | . |
| incavg | Input | Interval | No | | No | . | . |
| incmed | Input | Interval | No | | No | . | . |
| kids | Input | Interval | No | | No | . | . |
| lag | Input | Interval | No | | No | . | . |
| low | Input | Interval | No | | No | . | . |
| mdon | Input | Interval | No | | No | . | . |
| npro | Input | Interval | No | | No | . | . |
| ownd | Input | Binary | No | | No | . | . |
| region | Input | Nominal | No | | No | . | . |
| sex | Input | Binary | No | | No | . | . |
| wlth | Input | Ordinal | No | | No | . | . |

*Figure 11*

**Logistic Regression.** For the Logistic Regression, a stepwise regression with interactions and polynomials was selected with a maximum of 20 steps, 2$^{nd}$ degree polynomial, and an entry significance level of 0.1. The model's misclassification rate was 9% (Figure 12) on the validation data. From Figure 13, it can be interpreted that the stepwise function continued to improve the model's misclassification rate until the last step.

Fit Statistics

Target=donr Target Label=donr

| Fit Statistics | Statistics Label | Train | Validation |
|---|---|---|---|
| _AIC_ | Akaike's Information Criterion | 1953.18 | . |
| _ASE_ | Average Squared Error | 0.07 | 0.07 |
| _AVERR_ | Average Error Function | 0.23 | 0.22 |
| _DFE_ | Degrees of Freedom for Error | 4171.00 | . |
| _DFM_ | Model Degrees of Freedom | 29.00 | . |
| _DFT_ | Total Degrees of Freedom | 4200.00 | . |
| _DIV_ | Divisor for ASE | 8400.00 | 3604.00 |
| _ERR_ | Error Function | 1895.18 | 801.35 |
| _FPE_ | Final Prediction Error | 0.07 | . |
| _MAX_ | Maximum Absolute Error | 0.99 | 1.00 |
| _MSE_ | Mean Square Error | 0.07 | 0.07 |
| _NOBS_ | Sum of Frequencies | 4200.00 | 1802.00 |
| _NW_ | Number of Estimate Weights | 29.00 | . |
| _RASE_ | Root Average Sum of Squares | 0.26 | 0.26 |
| _RFPE_ | Root Final Prediction Error | 0.27 | . |
| _RMSE_ | Root Mean Squared Error | 0.26 | 0.26 |
| _SBC_ | Schwarz's Bayesian Criterion | 2137.12 | . |
| _SSE_ | Sum of Squared Errors | 585.37 | 242.01 |
| _SUMW_ | Sum of Case Weights Times Freq | 8400.00 | 3604.00 |
| _MISC_ | Misclassification Rate | 0.10 | 0.09 |

*Figure 12*

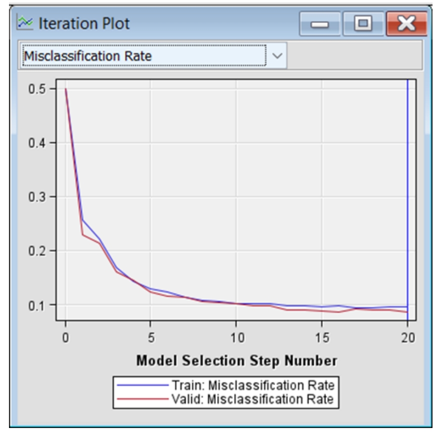



*Figure 13*

Figure 14 showcases the model's confusion matrix, which was used to calculate the sensitivity, specificity, and accuracy. Each calculation was as follows: Sensitivity(Validation) =827/899= 91.9%, Specificity (Validation)=816/903=90.4%, and Accuracy=1643/1802=91.2%.

Data Role=VALIDATE Target=donr Target Label=donr

| False Negative | True Negative | False Positive | True Positive |
|---|---|---|---|
| 72 | 816 | 87 | 827 |

*Figure 14*

**Neural Network.** The Neural Network first required data partitioning, after which a Transform Variable node was selected to perform standardization (rescaling) of the interval variables, following which the standardized inputs were connected to the neural networks' node. The model selection criteria was misclassification rate with 3 hidden units, training technique was back propagation, max time was 30 minutes, and the learning rate was set to 0.1. The model's misclassification rate was 12% (Figure 15) on the validation data. From the iteration plot (Figure 16), it can be noted that both training and validation misclassification rates decreased over time showing the model learned and improved with each iteration.

The consistency between the training and validation misclassification rates suggests there is no overfitting.

Fit Statistics

Target=donr Target Label=donr

| Fit Statistics | Statistics Label | Train | Validation |
|---|---|---|---|
| _DFT_ | Total Degrees of Freedom | 4200.00 | . |
| _DFE_ | Degrees of Freedom for Error | 4094.00 | . |
| _DFM_ | Model Degrees of Freedom | 106.00 | . |
| _NW_ | Number of Estimated Weights | 106.00 | . |
| _AIC_ | Akaike's Information Criterion | 2816.26 | . |
| _SBC_ | Schwarz's Bayesian Criterion | 3488.60 | . |
| _ASE_ | Average Squared Error | 0.09 | 0.09 |
| _MAX_ | Maximum Absolute Error | 0.96 | 0.95 |
| _DIV_ | Divisor for ASE | 8400.00 | 3604.00 |
| _NOBS_ | Sum of Frequencies | 4200.00 | 1802.00 |
| _RASE_ | Root Average Squared Error | 0.31 | 0.30 |
| _SSE_ | Sum of Squared Errors | 796.77 | 320.87 |
| _SUMW_ | Sum of Case Weights Times Freq | 8400.00 | 3604.00 |
| _FPE_ | Final Prediction Error | 0.10 | . |
| _MSE_ | Mean Squared Error | 0.10 | 0.09 |
| _RFPE_ | Root Final Prediction Error | 0.32 | . |
| _RMSE_ | Root Mean Squared Error | 0.31 | 0.30 |
| _AVERR_ | Average Error Function | 0.31 | 0.30 |
| _ERR_ | Error Function | 2604.26 | 1068.71 |
| _MISC_ | Misclassification Rate | 0.13 | 0.12 |
| _WRONG_ | Number of Wrong Classifications | 562.00 | 213.00 |

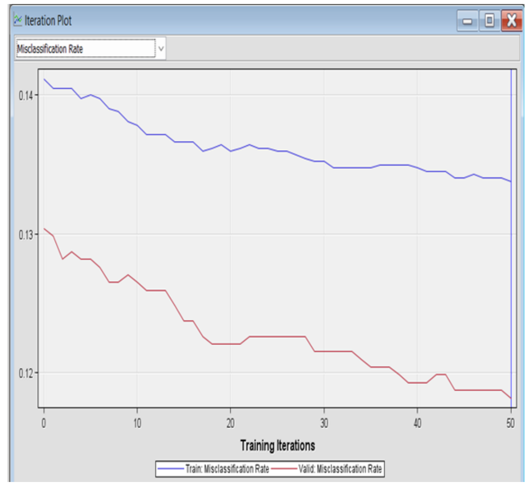



*Figure 15* *Figure 16*

Figure 17 showcases the model's confusion matrix, which was used to calculate the sensitivity, specificity, and accuracy. Each calculation was as follows; Sensitivity (Validation)= 819/899=91.1%, Specificity (Validation)=770/903=85.3%, and Accuracy=1589/1802=88.2%.

Data Role=VALIDATE Target=donr Target Label=donr

| False Negative | True Negative | False Positive | True Positive |
|---|---|---|---|
| 80 | 770 | 133 | 819 |

*Figure 17*

**Random Forest.** To create the Random Forest, an HPDM node was created and connected to the partition node. The validation misclassification rate was 11% (Figure 18). As seen in Figure 19, there is not a huge gap between the training and validation errors, indicating the model is not overfitting the data.

Fit Statistics

Target=donr Target Label=donr

| Fit Statistics | Statistics Label | Train | Validation |
|---|---|---|---|
| _ASE_ | Average Squared Error | 0.09 | 0.10 |
| _DIV_ | Divisor for ASE | 8400.00 | 3604.00 |
| _MAX_ | Maximum Absolute Error | 0.87 | 0.88 |
| _NOBS_ | Sum of Frequencies | 4200.00 | 1802.00 |
| _RASE_ | Root Average Squared Error | 0.30 | 0.31 |
| _SSE_ | Sum of Squared Errors | 770.09 | 343.93 |
| _DISF_ | Frequency of Classified Cases | 4200.00 | 1802.00 |
| _MISC_ | Misclassification Rate | 0.11 | 0.11 |
| _WRONG_ | Number of Wrong Classifications | 442.00 | 197.00 |

*Figure 18*

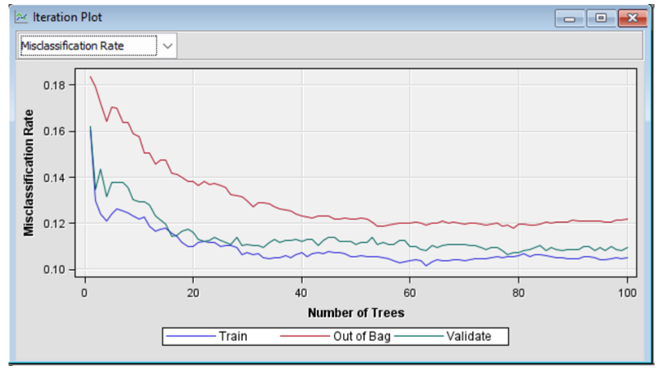



*Figure 19*

Figure 20 showcases the model's confusion matrix, which was used to calculate the sensitivity, specificity, and accuracy. Each calculation was as follows; Sensitivity (Validation) = 809/899 = 90.0%, Specificity (Validation) = 796/903 = 88.2%, and Accuracy = 1605/1802 = 89.1%.

Data Role=VALIDATE Target=donr Target Label=donr

| False Negative | True Negative | False Positive | True Positive |
|---|---|---|---|
| 90 | 796 | 107 | 809 |

*Figure 20*

## Regression

The target variable for regression here is damt. To accurately predict this variable the regression models KNN, regression trees, and multiple linear regression were chosen.

### *Regression Model Selection*

The first model employed for predicting the target variable damt was the parametric model K nearest Neighbors (KNN). Referred to as the lazy learner, this model does not learn from the training set immediately but it stores the dataset and at the time of prediction it performs an action on the dataset. The benefits to using the model include its ability to not make assumptions about the underlying data, its ease of implementation, and its ability to handle nonlinearity. The potential downside of this model includes the selection of the appropriate K value, it is susceptible to outliers, and it produces a complex prediction formula with slow prediction ability.

The second model selected was Regression Trees. The benefits to choosing this model include its ease to build, implement, and interpret, its lack of need for underlying assumptions or standardization, and its ability to handle missing data. The downsides of this model include its tendency to overfit the data and it requires large amounts of data, since the trees do not make assumptions.

The third and final model employed for predicting the regressor variable was multiple linear regression. The assumptions of this model include its ability to identify a relationship between the predictor and outcome variables, constant variance of residuals, independence of residuals, and lack of multicollinearity make it an appropriate choice for this dataset (LinkedIn, 2023). The weakness of this model comes in the form of difficulty in interpreting its results and misleading results if the assumption were to be violated.

***Regression Modelling Process***

Using the file import node the dataset was imported, and the target variable was set to damt for prediction while donr and id were rejected. Figure 21 showcases this in more detail. A data partition node was created and connected to divide the data into Training (70%) and Validation (30%).

Columns: Label Mining

| Name | Role | Level | Report | Order | Drop | Lower Limit | Upper Limit |
|---|---|---|---|---|---|---|---|
| damt | Target | Interval | No | | No | . | . |
| donr | Rejected | Binary | No | | No | . | . |
| gifa | Input | Interval | No | | No | . | . |
| gifdol | Input | Interval | No | | No | . | . |
| gifl | Input | Interval | No | | No | . | . |
| gifr | Input | Interval | No | | No | . | . |
| hv | Input | Interval | No | | No | . | . |
| ID | Rejected | Nominal | No | | No | . | . |
| inc | Input | Ordinal | No | | No | . | . |
| incavg | Input | Interval | No | | No | . | . |
| incmed | Input | Interval | No | | No | . | . |
| kids | Input | Interval | No | | No | . | . |
| lag | Input | Interval | No | | No | . | . |
| low | Input | Interval | No | | No | . | . |
| mdon | Input | Interval | No | | No | . | . |
| npro | Input | Interval | No | | No | . | . |
| ownd | Input | Binary | No | | No | . | . |
| region | Input | Nominal | No | | No | . | . |
| sex | Input | Binary | No | | No | . | . |
| wlth | Input | Ordinal | No | | No | . | . |

*Figure 21*

**KNN.** The VARIABLEXPLORE node was created with the Target Model as R square. Figure 22 shows the combination of each variable to overall R Square in predicting damt. Kids are the most important predictor of damt as it explains 30% of the variance. The transform Variables node was selected for standardization of all interval variables to a common scale. Here damt was selected to none as the target variable is already standardized. The Model was built with K = 12 as the MBR nodes.

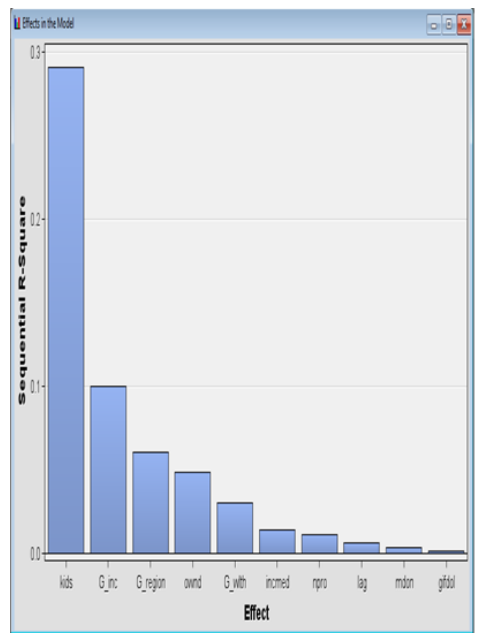



*Figure 22*

**Decision Trees.** The decision tree regressor was implemented with a significance level of 0.2, maximum branch of 6 and maximum depth of 20. It was then connected to the control point node which gives the control to halt before automatic pruning happens. The assessment mode was selected to prune the tree with absolute squared error (ASE) as the pruning measure. From Figure 23 it can been seen that the ASE was initially high for both training and validation data but as the number of leaves increased, both ASE decreased sharply. After 75 leaves, the validation ASE started to worsen even though the training ASE kept decreasing. The use of pruning aided in avoiding overfitting by selecting the optimal value of leaves to be 75.

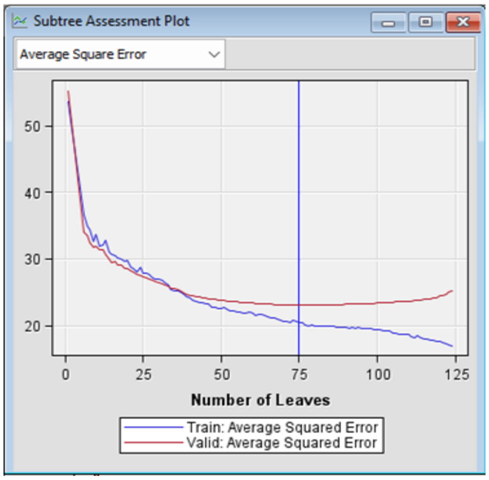



*Figure 23*

**Multiple Linear Regression.** An MLR model was built with the selection method as stepwise, an entry significance level of 0.1, and the number of steps set to 10.

## Model Evaluation

### Classification

The results from the classification model comparison were evaluated using the Model Comparison node within SAS EM, where model selection was based on the misclassification rate of the validation data. The logistic regression model with stepwise interactions and polynomial terms demonstrated the best performance (Figure 24), achieving the lowest misclassification rate on the validation set (0.097) and a similarly low rate on the training set (0.088). This consistency across the training and validation data suggests that the model generalizes well and is not overfitted. The next best performing model was the random forest model with a slightly higher misclassification rate of 0.105, indicating reduced accuracy in classifying donors. The neural network model, while more complex in structure, yielded the highest misclassification rate of

0.134 on the validation data, suggesting that it is less effective for predicting the target variable. Together, these results indicate that the logistic regression model is for identifying the target variable donr with accuracy and efficiency.

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

| Selected Model | Model Node | Model Description | Valid: Misclassification Rate | Train: Average Squared Error | Train: Misclassification Rate | Valid: Average Squared Error |
|---|---|---|---|---|---|---|
| Y | Reg | LR STEPWISE INTERACTIONS AND POLYNOMIALS | 0.08824 | 0.069687 | 0.09690 | 0.067151 |
| | HPDMForest | DONR=HP Forest | 0.10932 | 0.091677 | 0.10524 | 0.095429 |
| | Neural | DONR= NEURAL NTWRK 3 | 0.11820 | 0.094853 | 0.13381 | 0.089033 |

kierawingo@my.unt.edu as u64193493 Connected to SASApp - Logical Workspace Server (odaws01-usw2-2.oda.sas.com)

*Figure 24*

**Regression**

The results from the regression model comparison were evaluated using the Model Comparison node in SAS Enterprise Miner, with model selection based on the average squared error (ASE) from the validation dataset. The decision tree model, consisting of six branches and a depth of 20, performed the best showcasing (Figure 25) the lowest ASE (23.122) on the validation data and a comparably low ASE (20.591) on the training data. The closeness of these two metrics indicates that the model adequately captured the data's patterns without overfitting. The stepwise linear regression model was the next best with an ASE of 23.204 for the validation data but showed higher training error suggesting reduced generalization from training to validation data. All variations of the memory-based reasoning (MBR) models produced substantially higher validation ASE values, with the best-performing MBR variant (K = 12) yielding an ASE of 32.443. These findings indicate that the decision tree model is best suited to predict the target variable damt to determine donation amounts.

Fit Statistics
Model Selection based on Valid: Average Squared Error (_VASE_)

| Selected Model | Model Node | Model Description | Valid: Average Squared Error | Train: Average Squared Error |
|---|---|---|---|---|
| Y | Tree | 6 Branch 20 deep | 23.1226 | 20.5907 |
| | Reg2 | Amount- Stepwise | 23.2047 | 23.2170 |
| | MBR4 | MBR K = 12 | 32.4425 | 30.3668 |
| | MBR3 | MBR k= 11 | 32.7341 | 30.2215 |
| | MBR12 | MBR k = 9 | 32.8496 | 29.4675 |
| | MBR2 | MBR k = 10 | 32.9220 | 29.8596 |
| | MBR11 | MBR K = 8 | 33.5459 | 29.0488 |
| | MBR10 | MBR k = 7 | 33.9137 | 28.4723 |
| | MBR9 | MBR K= 6 | 34.4821 | 27.8970 |
| | MBR8 | MBR k =5 | 35.5132 | 27.1451 |
| | MBR7 | MBR K= 4 | 36.1005 | 25.7148 |
| | MBR6 | MBR k= 3 | 37.1099 | 23.7597 |
| | MBR5 | MBR k= 2 | 40.3045 | 21.4528 |
| | MBR | MBR k =1 | 45.3634 | 15.9244 |

kierawingo@my.unt.edu as u64193493 Connected to SASApp - Logical Workspace Server (odaws01-usw2-2.oda.sas.com)

*Figure 25*

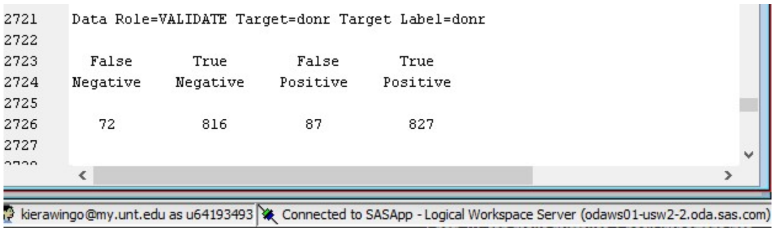
## Model Deployment

### Classification

For the deployment phase, the Score node was used to apply the best-performing classification model to the unlabeled score data (nonprofit_score.xlsx). This Score node automatically selected the stepwise logistic regression model based on its lowest misclassification rate on the validation data. The logistic regression model was then applied to generate predicted donor classifications for each observation in the new dataset.

To determine the expected profit from mailing based on model predictions, individuals who were classified as likely donors (I_donr = 1) were selected for targeted solicitation. Using the confusion matrix (Figure 26) from the validation dataset, the logistic regression model showed a precision of 90.3%, meaning that 827 of the 914 individuals predicted to be donors were actual donors. Based on this precision, the prior knowledge that each mailer cost $2.00 to send, and the average donation of $14.05, we can expect the profit per mailer to be calculated as follows:

$$\text{Expected Profit per Mailing} = (14.50 \times 0.903) - 2 = \$11.12$$



Data Role=VALIDATE Target=donr Target Label=donr

| False Negative | True Negative | False Positive | True Positive |
|---|---|---|---|
| 72 | 816 | 87 | 827 |



*Figure 26*

This projected profit represents a significant improvement over the baseline mass mailing strategy, which produced an expected loss of $0.55 per mailing due to the low 10% response rate. By applying the model to score new contacts and mailing only to those predicted as likely donors, the nonprofit organization can more effectively allocate its resources and improve the cost-effectiveness of its campaign efforts.

From a business perspective, this model proves to be a useful tool for devising efficient and effective donation strategies. The ability to focus on individuals with a high likelihood of donating will ultimately reduce costs and increase donation amounts. The Score node's output can be utilized to generate direct mailing lists and support data-driven decisions for future marketing campaigns. As a result, this model supports the nonprofit's goal of enhancing donor engagement while maintaining financial efficiency in its outreach efforts.

**Regression**

In addition to identifying likely donors, the client requested an estimate of expected donation amounts to further refine mailing decisions. The Score node was used to apply the best-performing regression model, selected by Model Comparison node, to the new unlabeled dataset. The decision tree model with six branches and a depth of 20 produced the lowest average squared error on validation data and was used to generate predicted donation amounts (P_damt) for individuals classified as likely donors (I_donr = 1). Using the scored data output, the average donation amount was found by filtering all rows for I_donr = 1, then averaging the P_damt values for rows equal to I_donr = 1. The resulting value ($11.13) and the cost to send a mailer ($2.00), can be used to determine the average donation amount as follows:

$$\text{Average Predicted Donation} = 11.13 - 2.00 = \$9.13$$

Where the classification models resulted in an average expected profit of $11.12, this average donation amount of $9.13 further refines the expected amount by utilizing a real average (11.13) from the data created by only looking at actual donor observations. Applied to our calculation, the resulting average donation amount of $9.13 is a data-driven result that is more accurate than averaging all values. Utilizing this targeted approach over the mass mailing strategy, which produces a loss of $0.55 per mailer, the organization can prioritize donors based not only on likelihood to give, but also on expected donation size. This strategy supports cost-effective fundraising and enables informed resource allocation for future campaigns.

## Conclusion

We passed along our results and recommendations for review after compiling all the data and modeling outcomes. We were able to find several meaningful indicators of highest and lowest potential donors. For example, prospects with middle or lower home values are more prone to donate. Region also seems to have an influence on who is more likely to donate. Unsurprisingly, high wealth people are also more likely to donate.

After piecing the whole puzzle together, we predict that by better targeting potential donors and removing less likely donors from their mailing list, the organization stands to make quite a profit. Using the classification model to estimate profit, we found that if they keep the cost per mailing at $2.00, they could stand to make a profit per mailer of $11.12, compared to their current loss of $0.55 each. This could also increase their response rate from 10% to 90% -- a substantial improvement in engagement. Using the regression model to predict average gift amount per donor, we found that this amount could be increased to $11.13, which still nets $9.13 per donor after factoring in the cost per mailing.

With these new data-driven predictions and estimates, we believe that the nonprofit will be able to make their fundraising campaigns more efficient and effective by better targeting the prospects most likely to donate.

## References

GeeksforGeeks. (2025, January 2). *Advantages and disadvantages of logistic regression*. https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/

LinkedIn. (2023, February 15). *What are the benefits and drawbacks of using stepwise methods for variable selection in multiple regression?* https://www.linkedin.com/advice/0/what-benefits-drawbacks-using-stepwise

## Appendix A

| **Project Part** | **Work Done By** | **Drafted By** | **Edited By** |
|---|---|---|---|
| Introduction | Ashley Chesser | Ashley Chesser | Kiera Wingo |
| EDA | Yitian Liu | Yitian Liu | Ashley Chesser, Kiera Wingo |
| Data Preparation | Yitian Liu | Yitian Liu | Ashley Chesser, Kiera Wingo |
| Modeling - Classification | Sana Ambreen | Sana Ambreen | Ashley Chesser, Kiera Wingo |
| Modeling - Regression | Sana Ambreen | Sana Ambreen | Ashley Chesser, Kiera Wingo |
| Model Evaluation | Sana Ambreen | Sana Ambreen | Ashley Chesser, Kiera Wingo |
| Model Deployment | Kiera Wingo | Kiera Wingo | Ashley Chesser, Kiera Wingo |
| Conclusion | Ashley Chesser | Ashley Chesser | Ashley Chesser |