MA478 TEE Report

CDT Joshua L. Blackmon

1 Research Question

A charity organization has essentially provided us with two requests to try to complete:

1. *Provide a model that will maximize the expected net profit.*
2. *Provide a model that will predict the expected gift amounts from those that donate.*

This is in an effort to fix the overall response rate based on the and maintain the most cost-effective way to mail everyone.

2 Data

2.1 Data Exploration/Preparation

The data we were given includes all that are in Figure 1 and 2.

ID: unique donor number [Do NOT use this as a predictor variable in any models]
REG1, REG2, REG3, REG4: Region (There are five geographic regions; only four are needed for analysis since if a potential donor falls into none of the four he or she must be in the other region. Inclusion of all five indicator variables would be redundant and cause some modeling techniques to fail. A "1" indicates the potential donor belongs to this region.)
HOME: (1 = homeowner, 0 = not a homeowner)
CHLD: Number of children
HINC: Household income (7 categories)
GENF: Gender (0 = Male, 1 = Female)
WRAT: 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.)
AVHV: Average Home Value in potential donor's neighborhood in $ thousands
INCM: Median Family Income in potential donor's neighborhood in $ thousands
INCA: Average Family Income in potential donor's neighborhood in $ thousands
PLOW: Percent categorized as "low income" in potential donor's neighborhood
NPRO: Lifetime number of promotions received to date
TGIF: Dollar amount of lifetime gifts to date
LGIF: Dollar amount of largest gift to date
RGIF: Dollar amount of most recent gift
TDON: Number of months since last donation
TLAG: Number of months between first and second gift
AGIF: Average dollar amount of gifts to date
DONR: Classification Response Variable (1 = Donor, 0 = Non-donor)
DAMT: Prediction Response Variable (Donation Amount in $).

Figure 1: Variables given in Charity Dataset.

Immediately, we notice a few things:

- REG1, REG2, REG3, and REG4 are all used to identify location of the potential donor.
- WRAT is an organized category.
- AVHV, INCM, and INCA are all in the thousands.

Regarding the Region used, we chose to merge them all into one variable that has 5 factors being each region. It may make the model faster and help ascertain a better prediction outcome.

For WRAT, since it is variable that is in order, we decided to ascribe an order to it correlating positively to the value. As 9 is the highest wealth, 9 is also the highest order in the variable.

Finally, for the three economic variables, we sought to explore the state at which the variables are impacted by each other. In Figure 2, you can see the correlation heatmap for all present quantitative variables to see how much they overlap.

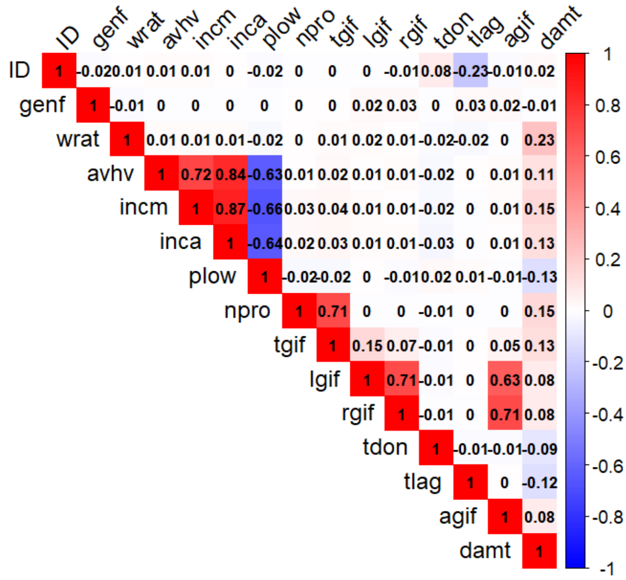

Figure 2: Correlation Heatmap with each Quantitative Variable

Figure 2 shows that the three are all correlated positively to some higher degree but this also show how the percentage of low income is also related to the three negatively. Anectdotally this makes sense as those with similar economic statuses tend to live near each other.

In Figure 3, we can see the direct degree that AVHV, INCM, and INCA are all correlated. This also shows how many outliers are present in the dataset regarding the data. For this reason, we elected to raise them to their log to minimize the impact of all outliers. This is in order to make the model more accurate.

Through further exploration, there was not missing data so we had no reason to impute or add anything from that.

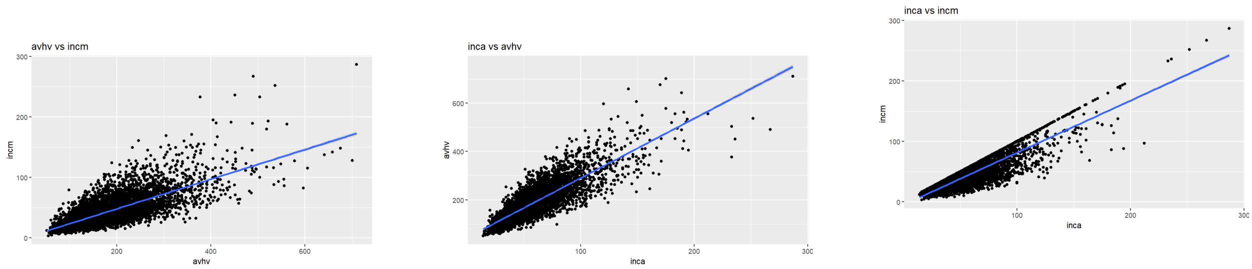

Figure 3: Overall caption for the group of images

3 Model Performance and Results

For the 3 models used, we settled on one full binomial model, one binomial model that uses step regression based on AIC to determine the right variables and one that used the same variables but through quasi-binomial. Model 2 and 3 had the same results regardless so my best model but Model 2 is easier to interpret. For this reason, we chose Model 2 as our model of choice.

Based on this model we will be able to mail 1433 of our highest posterior probabilities.

For the models used for the Amount Donated, we settled on our linear model using the stepwise function that gave us the lowest MSE: 89.79938 with the standard deviation of 3.451564.

4 Conclusions

The actual error rate of the model is quite high due to a lack of standardization which is one limit of the model. We also couldn't account for the effects that couldn't be covered (Location, Space, etc.) Outside of that, this is the best model possible.

5 Appendix A: Code Used

````
---
title: "TEE MA478 Blackmon"
author: "CDT Joshua Blackmon"
date: "`r Sys.Date()`"
output: word_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

```{r}
library(CARBayesST)
library(lmtest)
library(rstan)
library(Matrix)
library(dplyr)
library(tidyverse)
library(lubridate)
library(knitr)
library("spdep")
library("lme4")
library("spatstat")
library("sp")
# library("maptools")
library("lattice")
library(caTools)
library(pROC)
library(ggplot2)
library(MASS)
library(pscl)
library(nnet)
library(geepack)
library(INLA)
library(GGally)
library(MASS)
library(dplyr)
library(ggplot2)
library(lme4)
library(corrplot)

```


```{r}

charity_test <- read.csv(file.choose())
charity_overall <- read.csv(file.choose())

```

<!-- Split the data into training and validation based on if the 'part' column
    says train or valid. -->
<!-- ```{r} -->
````

```
<!-- charity_train <- charity_overall[charity_overall$part == "train",] -->
<!-- charity_valid <- charity_overall[charity_overall$part == "valid",] -->

<!-- ``` -->
The purpose of the train and validate splits are to train the model on the training data and then validate the model on the validation data. This is done to ensure that the model is not overfitting the training data and is generalizable to new data.

```{r}

summary(charity_overall)

#Make wrat column ordinal categorical since it is a scale of 0-9, 9 being the highest.

charity_overall$wrat <- ordered(charity_overall$wrat, levels = c("0","1", "2", "3", "4", "5", "6", "7", "8", "9"))
charity_test$wrat <- ordered(charity_test$wrat, levels = c("0","1", "2", "3", "4", "5", "6", "7", "8", "9"))

# Make home, chld, and hinc into Categorical Variables


charity_overall$home <- as.factor(charity_overall$home)
charity_overall$chld <- as.factor(charity_overall$chld)
charity_overall$hinc <- as.factor(charity_overall$hinc)

charity_test$home <- as.factor(charity_test$home)
charity_test$chld <- as.factor(charity_test$chld)
charity_test$hinc <- as.factor(charity_test$hinc)

charity_overall$donr <- as.factor(charity_overall$donr)
charity_overall$damt <- as.numeric(charity_overall$damt)

```
reg1, reg2,reg3, and reg4 are 4 different columns that are all categorical variables that represent 5 different regions. If there is a 1 in reg1, then a new column named region will say 1, if there is a 1 in reg2, then region will say 2, and so on. If there is a 0 in all of the reg columns, then the region will be 5.

```{r}

charity_overall <- charity_overall %>%
  pivot_longer(cols = starts_with("reg"), names_to = "region", values_to = "region_value") %>%
  group_by(ID) %>%
  mutate(region = ifelse(region_value == 1, region, "reg5")) %>%
  select(-region_value) %>%
  mutate(region = gsub("reg", "", region))

charity_overall <- charity_overall %>% group_by(ID) %>% filter(region == min(region))

```
```

````
charity_overall <- charity_overall %>% distinct(ID, .keep_all = TRUE)


charity_test <- charity_test %>%
  pivot_longer(cols = starts_with("reg"), names_to = "region", values_to = "region_value") %>%
  group_by(ID) %>%
  mutate(region = ifelse(region_value == 1, region, "reg5")) %>%
  select(-region_value) %>%
  mutate(region = gsub("reg", "", region))

charity_test <- charity_test %>% group_by(ID) %>% filter(region == min(region))

charity_test <- charity_test %>% distinct(ID, .keep_all = TRUE)

charity_overall$region <- as.factor(charity_overall$region)
charity_test$region <- as.factor(charity_test$region)

summary(charity_overall)

```


Find if there are any missing variables in the data set.

```{r}

sum(is.na(charity_overall))


sum(is.na(charity_test))

```

No missing data so no need to impute. Time to check correlation between variables.



```{r}

#create a df from the training set that only includes the columns that are quantitative

charity_overall_quant <- charity_overall[, sapply(charity_overall, is.numeric)]
corr_matrix <- cor(charity_overall_quant)
corrplot(corr_matrix, method = "color", type = "upper",
         tl.col = "black", tl.srt = 45,
         addCoef.col = "black", number.cex = 0.7,
         col = colorRampPalette(c("blue", "white", "red"))(200))


```
Here are the plots for each significant correlated variable.

```{r}

````

```
#Plot for lgif and agif

ggplot(charity_overall, aes(x = lgif, y = agif)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "lgif vs agif", x = "lgif", y = "agif")

#Plot for rgif and agif

ggplot(charity_overall, aes(x = rgif, y = agif)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "rgif vs agif", x = "rgif", y = "agif")

#plot for avhv and incm, inca and avhv, and inca and incm all on one plot

ggplot(charity_overall, aes(x = avhv, y = incm)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "avhv vs incm", x = "avhv", y = "incm")

ggplot(charity_overall, aes(x = inca, y = avhv)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "inca vs avhv", x = "inca", y = "avhv")

ggplot(charity_overall, aes(x = inca, y = incm)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "inca vs incm", x = "inca", y = "incm")

```
The plots show many outliers as well for the variables. We will attempt to minimize the impact of outliers through standardizing the income and value variables.

```{r}

charity_overall$incm <- log(charity_overall$incm)
charity_overall$inca <- log(charity_overall$inca)
charity_overall$avhv <- log(charity_overall$avhv)

charity_test$incm <- log(charity_test$incm)
charity_test$inca <- log(charity_test$inca)
charity_test$avhv <- log(charity_test$avhv)


```


We seek to determine the expected profit from each mailing. We k
Log Regression with everything in the model

```{r}

```

````
#split overall data into train and validation sets based on if the part column says train or valid

charity_train <- charity_overall[charity_overall$part == "train",]
charity_valid <- charity_overall[charity_overall$part == "valid",]
```

```{r}

x_train <- charity_train[, !(names(charity_train) %in% c("damt", "part",'ID','donr'))]

c_train <- charity_train[,(names(charity_train) %in% c('donr'))]# donr
n_train_c <- 3984 # 3984

#damt for rows with donr=1

y_train <- charity_train$damt[charity_train$donr == 1]# damt for observations with donr=1
n_train_y <- 1995

```

```{r}


x_valid <- charity_valid[, !(names(charity_valid) %in% c("damt", "part",'ID','donr'))]

c_valid <- charity_valid[,(names(charity_valid) %in% c('donr'))]# donr
n_valid_c <- 2018

#damt for rows with donr=1

y_valid <- charity_valid$damt[charity_valid$donr == 1]# damt for observations with donr=1
n_valid_y <- 999

```

```{r}

n_test <- 2007
x_test <- charity_test[, !(names(charity_test) %in% c("damt", "part",'ID','donr'))]

```

```{r}

x_train_mean <- apply(x.train, 2, mean)
x_train_sd <- apply(x.train, 2, sd)
x_train_std <- t((t(x.train)-x.train.mean)/x.train.sd) # standardize to have zero mean and unit sd
apply(x_train_std, 2, mean) # check zero mean
apply(x_train_std, 2, sd) # check unit sd
````

```
data_train <- data.frame(x.train.std, donr=c.train) # to classify donr
data_train_y <- data.frame(x.train.std[c.train==1,], damt=y.train) # to predict damt when donr=1

x_valid <- t((t(x.valid)-x.train.mean)/x.train.sd) # standardize using training mean and sd
data_valid <- data.frame(x.valid.std, donr=c.valid) # to classify donr
data_valid_y <- data.frame(x.valid.std[c.valid==1,], damt=y.valid) # to predict damt when donr=1

x_test<- t((t(x.test)-x.train.mean)/x.train.sd) # standardize using training mean and sd
data_test <- data.frame(x.test.std)

```

Cycling through the best binomial model based on Accuracy

```{r}

best_model <- NULL
best_mse <- Inf

data.train <- data.frame(x_train, donr = c_train)
data.train$donr <- as.factor(data.train$donr)

# create full model

full_model <- glm(donr ~ ., data = data.train, family = binomial(link = "logit"))
summary(full_model)

#model2 based on AIC

#model2 <- step(full_model, direction = "both", trace = 0)
model2 <- glm(formula = donr ~ home + chld + hinc + wrat + incm + plow +
    npro + tgif + tdon + tlag + region, family = binomial(link = "logit"),
    data = data.train)
summary(model2)

#model3 using quasi-binomial

model3 <- glm(formula = donr ~ home + chld + hinc + wrat + incm + plow +
    npro + tgif + tdon + tlag + region, family = quasibinomial(link = "logit"),
    data = data.train)
summary(model3)

```
```{r}

# Calculate ordered profit function using average donation = $14.50 and mailing cost = $2

# Calculate profit for each model

pred1 <- predict(full_model, newdata = data_vlau, type = "response")
profit.log1 <- cumsum(14.5*c.valid[order(pred1, decreasing=T)]-2)
```

````
plot(profit.log1)

pred2 <- predict(model2, newdata = x_valid, type = "response")
profit.log2 <- cumsum(14.5*c.valid[order(pred2, decreasing=T)]-2)
plot(profit.log2)

pred3 <- predict(model3, newdata = x_valid, type = "response")
profit.log3 <- cumsum(14.5*c.valid[order(pred3, decreasing=T)]-2)
plot(profit.log3)

```
Report the number for maximum profit

```{r}
which.max(profit.log1)
which.max(profit.log2)
which.max(profit.log3)
max(profit.log1)
max(profit.log2)
max(profit.log3)

```

```{r}
cutoff.log <- sort(pred2, decreasing=T)[which.max(profit.log2)+1] # set cutoff based on n.mail.valid
chat.valid.log <- ifelse(pred2>cutoff.log, 1, 0) # mail to everyone above the cutoff
table(chat.valid.log, c.valid)

```


```{r}


post.test <- predict(model2, x_test, type="response") # post probs for test data

# Oversampling adjustment for calculating number of mailings for test set
n.mail.valid <- which.max(profit.log1)
tr.rate <- .1 # typical response rate is .1
vr.rate <- .5 # whereas validation response rate is .5
adj.test.1 <- (n.mail.valid/n.valid.c)/(vr.rate/tr.rate) # adjustment for mail yes
adj.test.0 <- ((n.valid.c-n.mail.valid)/n.valid.c)/((1-vr.rate)/(1-tr.rate)) # adjustment for mail no
adj.test <- adj.test.1/(adj.test.1+adj.test.0) # scale into a proportion
n.mail.test <- round(n.test*adj.test, 0) # calculate number of mailings for test set

cutoff.test <- sort(post.test, decreasing=T)[n.mail.test+1] # set cutoff based on n.mail.test
chat.test <- ifelse(post.test>cutoff.test, 1, 0) # mail to everyone above the cutoff
table(chat.test)

```
````
````

````
```{r}

data_train_y <- charity_train[, !(names(charity_train) %in% c("part",'ID','donr'))]

model1 <- lm(damt ~ ., data_train_y)

summary(model1)

#choose model 2 based on AIC

model2 <- step(model1, direction = "both", trace = 0)
summary(model2)

#choose model 3 using stepwise selection

model3 <- step(model1, direction = "both", trace = 0, k = log(n_train_y))
summary(model3)

```

```{r}
#compare the models using the validation set

data_valid_y <- charity_valid[, !(names(charity_train) %in% c("part",'ID','donr'))]

pred.valid1 <- predict(model1, newdata = data_valid_y) # validation predictions
mean((y_valid - pred.valid1)^2) # mean squared error
# 1.867523
sd((y_valid - pred.valid1)^2)/sqrt(n_valid_y)

pred.valid2 <- predict(model2, newdata = data_valid_y) # validation predictions
mean((y_valid - pred.valid2)^2) # mean squared error
sd((y_valid - pred.valid2)^2)/sqrt(n_valid_y)

pred.valid3 <- predict(model3, newdata = data_valid_y) # validation predictions
mean((y_valid - pred.valid3)^2) # mean squared error
sd((y_valid - pred.valid3)^2)/sqrt(n_valid_y)

```

```{r}
yhat.test <- predict(model3, newdata = charity_test) # test predictions)

length(chat.test) # check length = 2007
length(yhat.test) # check length = 2007
chat.test[1:10] # check this consists of 0s and 1s
yhat.test[1:10]

ip <- data.frame(ID=data.test$ID, donr=chat.test, damt=yhat.test)
ip$damt[ip$donr == 0] <- 0

```
```{r}
````
````

````
idx.donors <- ip[(ip$donr > 0), ]
num_donors <- nrow(idx.donors)
est.profits <- sum(idx.donors$damt) - 2*num_donors
round(est.profits,2)

```


```{r}

submit <- data.frame(ID=ip$ID, damt=ip$damt) # data frame with two variables: ID and DAMT
write.csv(submit, file="Blackmon_submission.csv", row.names=FALSE)
```



```{r}

c.valid <- charity_valid[, (names(charity_valid) %in% c("donr"))]

profit.log1 <- cumsum(14.5*c.valid[order(log_donr_pred, decreasing=T)]-2)


```
````
````