Statistical Methods for Discrete Response, Time Series, and Panel Data (W271): Lab 2

Eric Yang, Samir Datta, Carlos Castro

OCTOBER 22, 2017

Introduction

Here we present the results of our analysis on contributions data for the university foundation, where our goal is to utilize the data available to predict who are likely to donate in the future and an idea of the magnitude of such donation.

The dataset includes, for each record, information such as:

- Donation amounts for the last 4 years
- Whether they attended the contribution events between 2012 and 2015
- Year of graduation
- Marital status, Gender
- Major of studies
- Year of graduation

Given that the goal of this study is to be able to predict who are likely to donate and the magnitude of that donation, we build a model focusing on its predictive power rather than its explanatory power. To build our model, in the following sections we thoroughly analyze the data, afterwards consider both multinomial and ordinal models and conduct statistical analysis on them to choose the one most fit for the required predictions.

As part of priorizing prediction over explanation in our models, we used a machine learning approach to testing and selecting models. We split our data in train and test set, to validate the predictions. In addition, to select the most powerful yet parsimious model predictors, we used statistical analysis, a thorough exploratory data analysis and domain knowledge.

We used confusion matrices on our models to understand exactly the strengths and weaknesses of our models' predictive power.

Finally, we selected an ordinal model that is based off the following predictors:

- AttendenceEvent: Whether they attended the donor events between 2012 and 2015
- Gender
- NextDegreeBinary: Binary variable of whether they would pursue another degree
- as.ordered(YearsSinceGrad): ordinal variable based off years since graduation (it has only 4 levels)
- log(meandonation+1) * gaveLastYear: Interaction between log of mean donation for the individual between 2012 and 2015 plus one, and whether it gave money last year or not

The sections to follow will include a thorough data analysis, starting with univariate analysis and then observing interactions with our output variable. After that, we'll do a model selection section where we explain our process towards the final model, followed by final remarks on our investigation.

Data Analysis

Data Loading

Note that we use two special libraries in this investigation:

- GGally: For parallel coordinate plots
- caret: For computing confusion matrices with thorough statistics

```
#loading packages and data
library(ggplot2)
library(ordinal)
```

```
## Warning: package 'ordinal' was built under R version 3.4.2
```

```
library(nnet)
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.4.2
```

```
## Loading required package: lattice
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 3.4.2
```

```
labdata <- read.csv('lab2data.csv')
```

Feature engineering

```
#split major into STEM and non-STEM
labdata$MajorType <- ifelse(labdata$Major=='Biology'|
                              labdata$Major=='Economics'|
                              labdata$Major=='Psychology'|
                              labdata$Major=='Physics'|
                              labdata$Major=='Chemistry'|
                              labdata$Major=='Mathematics'|
                              labdata$Major=='General Science-Chemistry'|
                              labdata$Major=='Economics-Business'|
                              labdata$Major=='General Science-Chemistry'|
                              labdata$Major=='Sociology-Anthropology'|
                              labdata$Major=='General Science-Psycho'|
                              labdata$Major=='General Science-Math'|
                              labdata$Major=='General Science-Biology'|
                              labdata$Major=='Computer Science'|
                              labdata$Major=='General Science'|
                              labdata$Major=='Mathematics-Physics'|
                              labdata$Major=='Economics-Regional Stds.'|
                              labdata$Major=='Zoology'|
                              labdata$Major=='Engineering'|
                              labdata$Major=='Sociology'|
                              labdata$Major=='Anthropology'|
                              labdata$Major=='General Science-Physics',
                            "STEM", "Non-STEM")

#create variable nextDegreeType to categorize the most common next degrees
```

```r
labdata$NextDegreeType <- ifelse(labdata$Next.Degree=='JD', 'JD',
                         ifelse(labdata$Next.Degree=='MA', 'MA',
                         ifelse(labdata$Next.Degree=='PHD', 'PHD',
                        ifelse(labdata$Next.Degree=='NDA', 'NDA',
                               ifelse(labdata$Next.Degree=='MS', 'MS',
                                      ifelse(labdata$Next.Degree=='MD', 'MD',
                                             ifelse(labdata$Next.Degree=='MBA', 'MBA',
         ifelse(labdata$Next.Degree=='NONE', 'NONE', 'Other'))))))))

#create simpler variable to represent if someone has an advanced degree or not
labdata$NextDegreeBinary <- ifelse(labdata$Next.Degree=='NONE', 0, 1)


#create buckets for all years
labdata$FY16cat <- cut(labdata$FY16Giving, c(0,1,100,250,500,200000), right=F)
labdata$FY15cat <- cut(labdata$FY15Giving, c(0,1,100,250,500,200000), right=F)
labdata$FY14cat <- cut(labdata$FY14Giving, c(0,1,100,250,500,200000), right=F)
labdata$FY13cat <- cut(labdata$FY13Giving, c(0,1,100,250,500,200000), right=F)
labdata$FY12cat <- cut(labdata$FY12Giving, c(0,1,100,250,500,200000), right=F)

#turn class year into years since grad to make interpretaion easier
labdata$YearsSinceGrad <- 2017 - labdata$Class.Year


#loop through data to get each person's mean donation over the past 4 years
#and how many of the past years they've donated
labdata$meandonation <- NA
labdata$nPastYears <- NA
for (i in c(1:1000)){
  labdata[i,]$meandonation<-mean(c(labdata[i,]$FY12Giving,
                                    labdata[i,]$FY13Giving,
                                    labdata[i,]$FY14Giving,
                                    labdata[i,]$FY15Giving))

  labdata[i,]$nPastYears <- sum(c(labdata[i,]$FY12Giving>0,
                                  labdata[i,]$FY13Giving>0,
                                  labdata[i,]$FY14Giving>0,
                                  labdata[i,]$FY15Giving>0))
}

#binary variable - have they donated before or not?
labdata$past_binary <- ifelse(labdata$meandonation == 0,0,1)

#did they donate last year?
labdata$donatedLastYear <- ifelse(labdata$FY15Giving==0, 0, 1)

#how many of the past 4 years, consecutively, did they donate?
#note that this will give a 0 for those that donated 2012-2014 but NOT 2015
#since we're asking for consecutive years
labdata$nPastConsecutiveYears <- ifelse(labdata$FY15Giving==0, 0,
                                        ifelse(labdata$FY14Giving==0,1,
                                               ifelse(labdata$FY13Giving==0,2,
                                                      ifelse(labdata$FY12Giving==0,3,4))))
```

```
#did they give in 2015 or not?
labdata$gaveLastYear <- ifelse(labdata$FY15Giving==0,0,1)
```

```
labdata$donatedLastYear <- ifelse(labdata$FY15Giving==0, 0, 1)

labdata$nPastConsecutiveYears <- ifelse(labdata$FY15Giving==0, 0,
                                        ifelse(labdata$FY14Giving==0,1,
                                               ifelse(labdata$FY13Giving==0,2,
                                                      ifelse(labdata$FY12Giving==0,3,4))))
labdata$gaveLastYear <- ifelse(labdata$FY15Giving==0,0,1)
```

EDA Univariate analysis

Gender

```
table(labdata$Gender)
```

```
##
##   F   M
## 505 495
```

Both male and female alumni are approximately evenly represented.

Years since graduation

```
table(labdata$YearsSinceGrad)
```

```
##
##   5  15  25  35  45
## 293 223 203 176 105
```

For the sake of an easier interpretation we transformed the varaible "Class.Year" into years since graduation by subtracting it from 2017. There are 5 unique values which reveals that this dataset polled alumni from classes 10 years apart. Younger alumni are more represented - alumni who graduated 5 years ago are represented almost 3 times as much as those who graduated 45 years ago. Already this appears to be an important variable to control for so that our model is not biased towards younger graduates. Furthermoer, due to the groups of the variable, we will want to treat this as an ordinal - instead of a continuous - variable.

Marital.Status

```
table(labdata$Marital.Status)
```

```
##
##   D   M   S   W
##  61 584 344  11
```

Most alumni are married, a good portion are single as well. The divorced and widowed group are sparsely represented.

Major

```
length(unique(labdata$Major))
```

```
## [1] 45
```

```
table(labdata$MajorType)
```

```
## 
## Non-STEM     STEM 
##      522      478
```

There are 45 different majors with varying levels of representation, including many with only one alumnus (for the sake of saving space we have decided to not show the full list). Because of that, we condensed this variable into STEM vs. Non-STEM, both of which appear to be approximately equally represented.

Next Degree

```
table(labdata$Next.Degree)
```

```
## 
##   AA   BA  BAE   BD  BFA   BN   BS  BSN   DC  DDS  DMD   DO  DO2   DP   JD 
##    1    4    1    1    1    2    2    3    1    1    1    2    1    1   90 
##  LLB  LLD   MA  MA2  MAE MALS  MAT  MBA  MCP   MD  MD2   ME  MFA  MHA   ML 
##    1    1  108    1    1    1   10   34    1   42    9   17   14    1    1 
##  MLS   MM  MPA  MPH   MS  MSM  MSW  NDA NONE  PHD  STM   TC UBDS UDDS  UMD 
##    9    1    6    4   53    1   11   58  378   78    1   22    6    4    6 
## UMDS UNKD 
##    2    6
```

```
table(labdata$NextDegreeBinary)
```

```
## 
##   0   1 
## 378 622
```

Like the Major variable, there is a variety of sparsely represented advanced degrees, so we chose to condense it into a binary variable - “None” vs. the rest. Interestingly, a considerable majority of alumni in this sample have an advanced degree, which could point to a sampling bias.

Attendance Event

```
table(labdata$AttendenceEvent)
```

```
## 
##   0   1 
## 395 605
```

A majority of alumni have attended alumni events between 2012 and 2015. This could also point to sampling bias - the dataset may come from alumni who were already more likely to donate than not.

Previous donations

```
table(labdata$FY12cat)
```

```
## 
##       [0,1)     [1,100)   [100,250)   [250,500) [500,2e+05) 
##         558         213         149          37          43
```

```
table(labdata$FY13cat)
```

```
## 
##       [0,1)     [1,100)   [100,250)   [250,500) [500,2e+05) 
##         513         247         143          54          43
```

```
table(labdata$FY14cat)
```

```
## 
##       [0,1)     [1,100)   [100,250)   [250,500) [500,2e+05) 
##         553         226         136          36          49
```

```
table(labdata$FY15cat)
```

```
## 
##       [0,1)     [1,100)   [100,250)   [250,500) [500,2e+05) 
##         567         199         138          36          60
```

From 2012 to 2015 the number of people in each donation category appears relativly stable. Higher donation categories have less alumni, with the exception of the highest category [500,2e+05) which has more than the next highest one in 3/4 years.

FY16 category

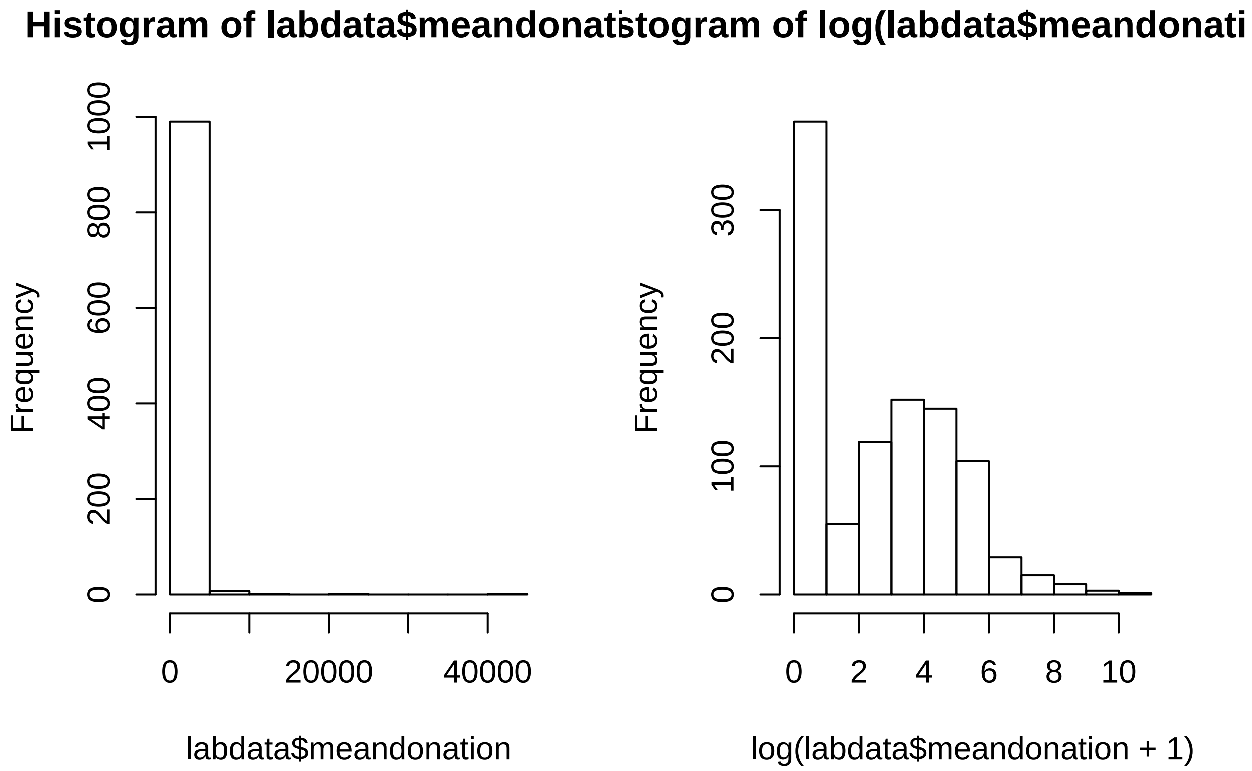
```
table(labdata$FY16cat)
```

```
## 
##       [0,1)     [1,100)   [100,250)   [250,500) [500,2e+05) 
##         586         173         143          39          59
```

The numbers for 2016 also look very similar to the previous years. This suggests that a large number of alumni stay in the same donation category from year to year, and that implementing information about previous years' donations will be crucial for our model's predictive ability. As we noticed before, the [250,500) category is very sparsely represented, which may make it hard to predict accurately.

Mean donation in the past

```
par(mfrow=c(1,2))
hist(labdata$meandonation)
hist(log(labdata$meandonation+1))
```

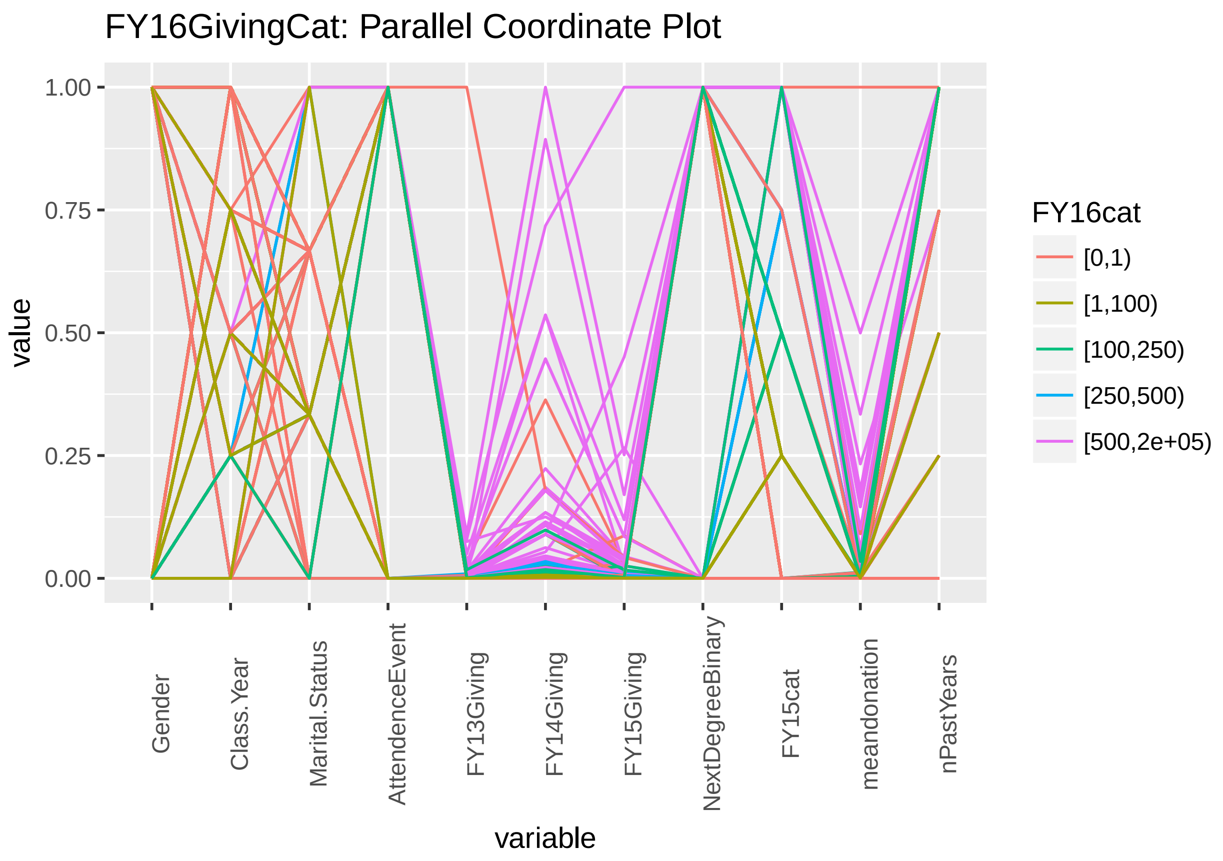


The variable mean donation, which represents each alumnus' mean donation from 2012-2015, has a large positive skew. Applying a log transformation (after adding 1, since the value 0 can't be log transformed) solves this to some extent, although a slight positive skew is still evident. A disproportionate number of alumni have a mean donation value of 0.

EDA Relationship between FY16cat and other variables

Parallel coordinate plot

```
ggparcoord(labdata, columns = c(2, 3, 4, 7, 9, 10, 11, 15, 17, 22, 23), groupColumn = "FY16cat", scale =
```

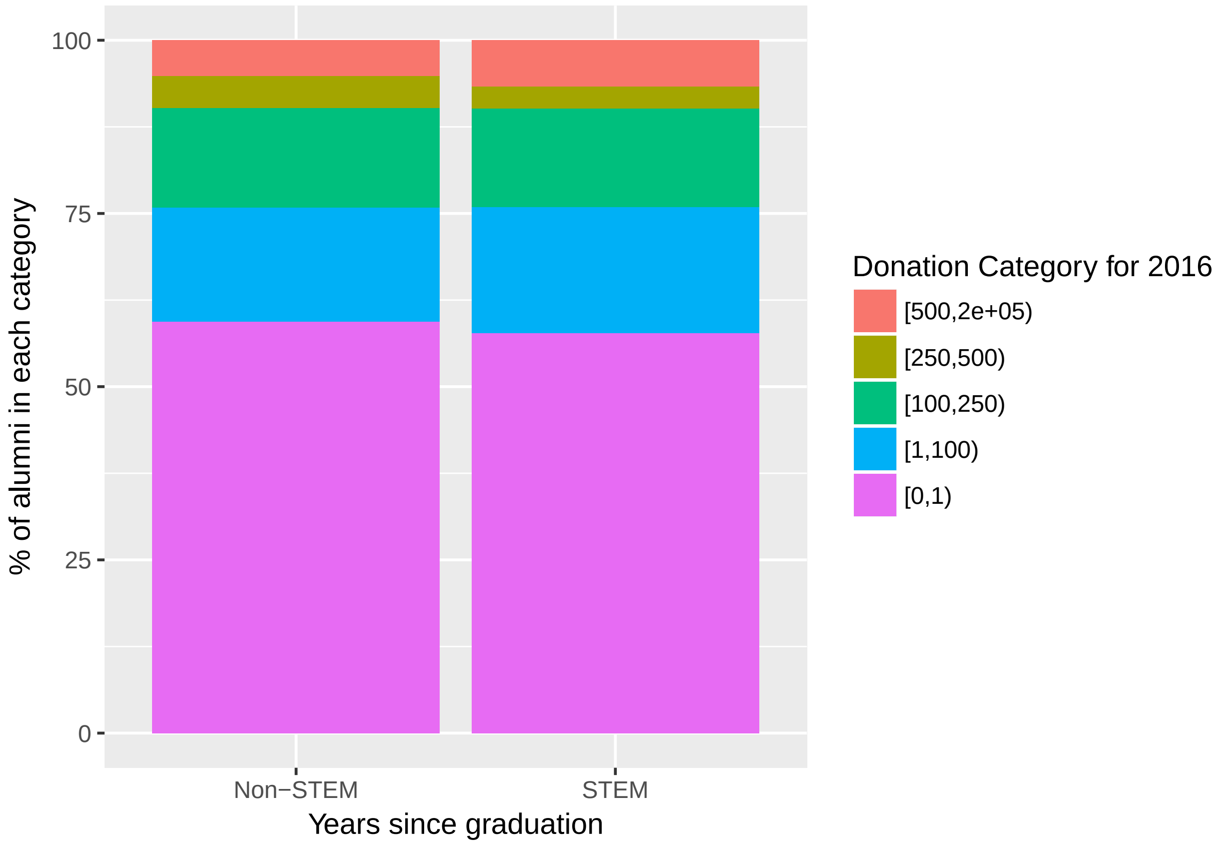


Major type - STEM vs. Non-STEM

```
labdata_counts <- with(labdata,
                       aggregate(MajorType,
                                 list(MajorType=MajorType),
                                 length))
labdata_agg <- with(labdata,
                    aggregate(MajorType, list(MajorType=MajorType,
                                              FY16cat=FY16cat),
                    length))
labdata_agg <- merge(labdata_agg, labdata_counts, by="MajorType")
labdata_agg <- setNames(labdata_agg, c("MajorType", "FY16cat", "Count", "TotalCount"))
labdata_agg$percent <- 100*labdata_agg$Count/labdata_agg$TotalCount
labdata_agg$FY16cat <- factor(labdata_agg$FY16cat,
      levels=c("[500,2e+05)","[250,500)","[100,250)","[1,100)","[0,1)"))

ggp <- ggplot(labdata_agg, aes(x=MajorType,y=percent))

ggp + geom_bar(stat="identity", aes(fill=FY16cat))+
  ylab("% of alumni in each category")+xlab("Years since graduation")+
  scale_fill_discrete(name="Donation Category for 2016")
```



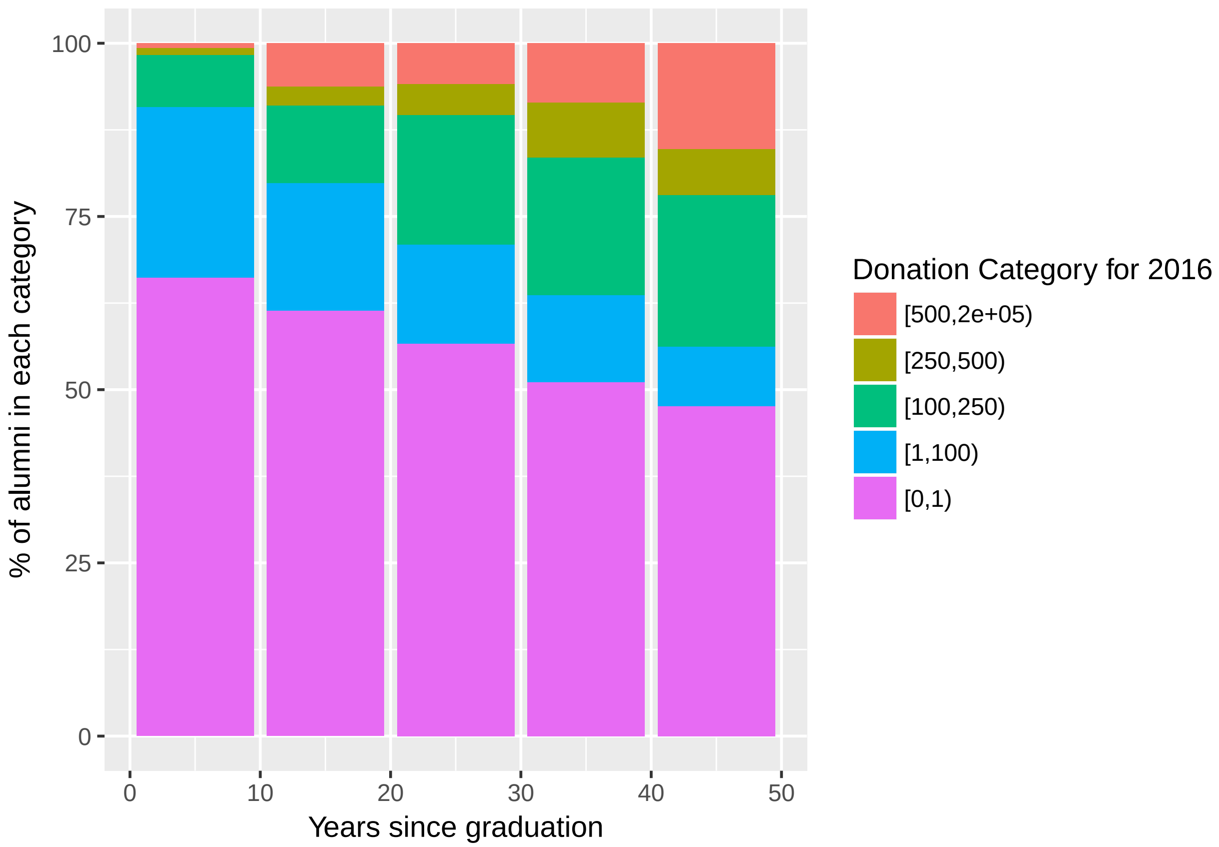
Major type does not seem to have an effect on the donation amount of an alumnnus, as the distribution of donation categories appears virtually identical regardless of whether they graduated with a STEM or non-STEM degree.

Next Degree (binary)

```
labdata_counts <- with(labdata,
                       aggregate(YearsSinceGrad,
                                 list(YearsSinceGrad=YearsSinceGrad),
                                 length))
labdata_agg <- with(labdata,
                    aggregate(YearsSinceGrad, list(YearsSinceGrad=YearsSinceGrad,
                                                    FY16cat=FY16cat),
                    length))
labdata_agg <- merge(labdata_agg, labdata_counts, by="YearsSinceGrad")
labdata_agg <- setNames(labdata_agg, c("YearsSinceGrad", "FY16cat", "Count", "TotalCount"))
labdata_agg$percent <- 100*labdata_agg$Count/labdata_agg$TotalCount
labdata_agg$FY16cat <- factor(labdata_agg$FY16cat,
      levels=c("[500,2e+05)","[250,500)","[100,250)","[1,100)","[0,1)"))

ggp <- ggplot(labdata_agg, aes(x=YearsSinceGrad,y=percent))

ggp + geom_bar(stat="identity", aes(fill=FY16cat))+
  ylab("% of alumni in each category")+xlab("Years since graduation")+
  scale_fill_discrete(name="Donation Category for 2016")
```



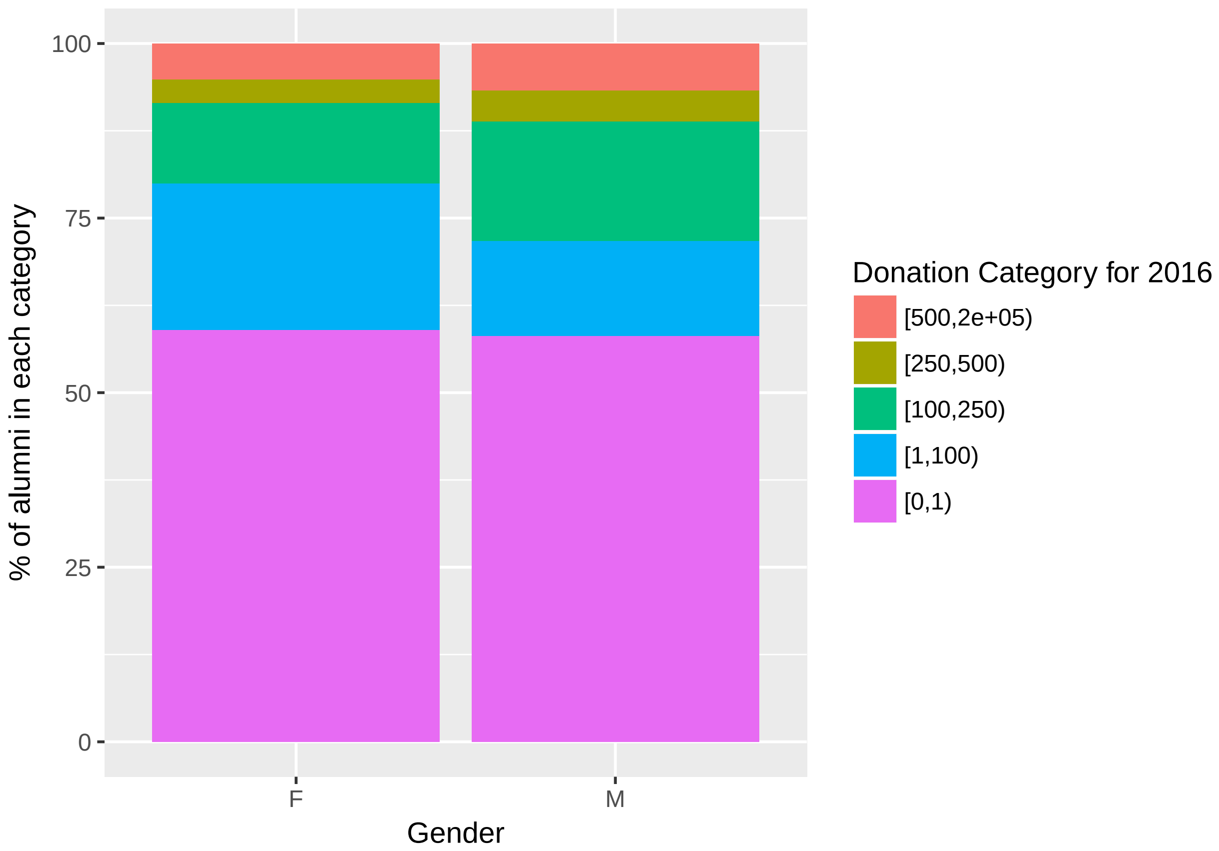
A clear ordinal relationship between years since grad and the amount donated in 2016 is shown in the violin plot, where those that graduated longer ago are more likely to not be in the [0,1] category and more likely to be in higher donation categories as well.

Gender

```
labdata_counts <- with(labdata,
                       aggregate(Gender,
                                 list(Gender=Gender),
                                 length))
labdata_agg <- with(labdata,
                    aggregate(Gender, list(Gender=Gender,
                                            FY16cat=FY16cat),
                    length))
labdata_agg <- merge(labdata_agg, labdata_counts, by="Gender")
labdata_agg <- setNames(labdata_agg, c("Gender", "FY16cat", "Count", "TotalCount"))
labdata_agg$percent <- 100*labdata_agg$Count/labdata_agg$TotalCount
labdata_agg$FY16cat <- factor(labdata_agg$FY16cat,
      levels=c("[500,2e+05)","[250,500)","[100,250)","[1,100)","[0,1)"))

ggp <- ggplot(labdata_agg, aes(x=Gender,y=percent))

ggp + geom_bar(stat="identity", aes(fill=FY16cat))+
  ylab("% of alumni in each category")+xlab("Gender")+
  scale_fill_discrete(name="Donation Category for 2016")
```



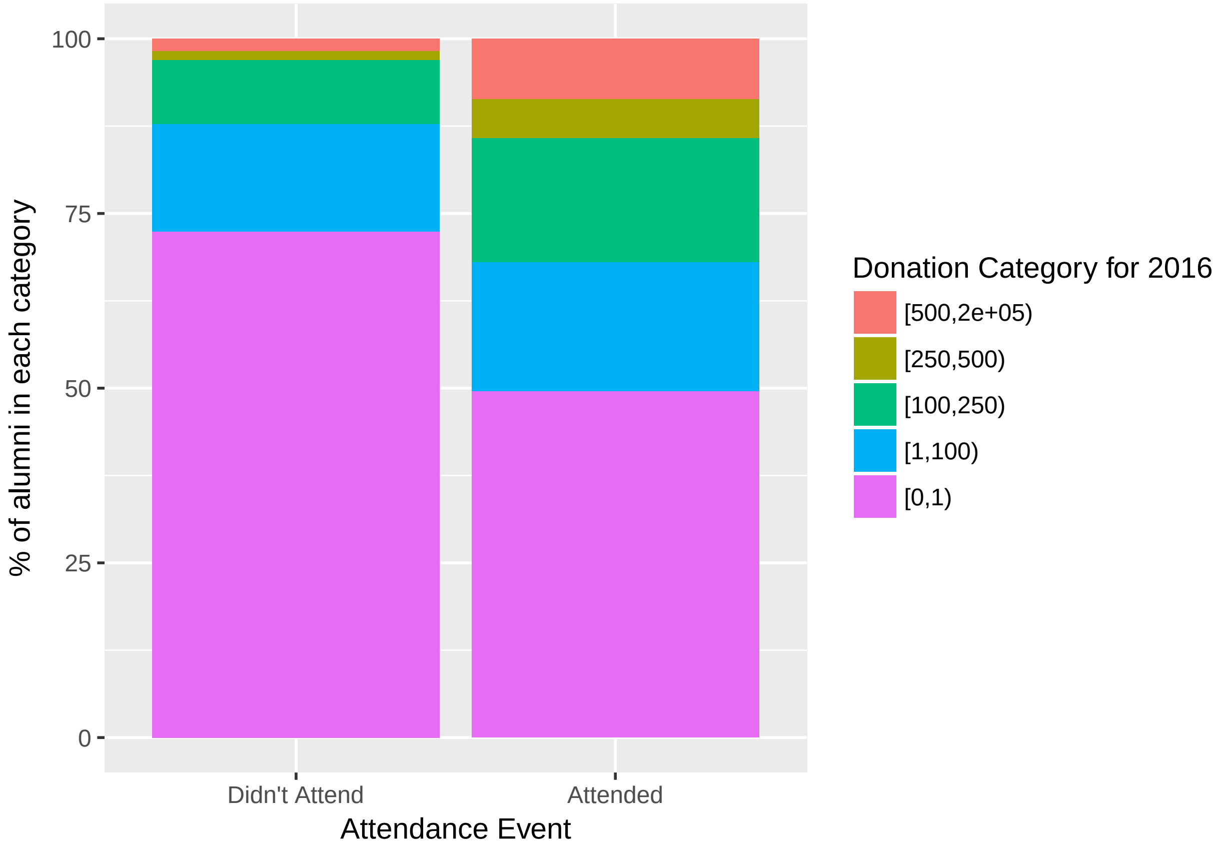
Men appear to be more likely to donate in the top 3 categories, while women appear to be more likely to donate in the [1,100) category. Interestingly, both men and women appear to be just as likely to donate nothing. This may suggest that gender would be more useful for a multinomial model instead of an ordinal model.

Attendance event

```
labdata_counts <- with(labdata,
                       aggregate(AttendenceEvent,
                                 list(AttendenceEvent=AttendenceEvent),
                                 length))
labdata_agg <- with(labdata,
                    aggregate(AttendenceEvent, list(AttendenceEvent=AttendenceEvent,
                                            FY16cat=FY16cat),
                    length))
labdata_agg <- merge(labdata_agg, labdata_counts, by="AttendenceEvent")
labdata_agg <- setNames(labdata_agg, c("AttendenceEvent", "FY16cat", "Count", "TotalCount"))
labdata_agg$percent <- 100*labdata_agg$Count/labdata_agg$TotalCount
labdata_agg$FY16cat <- factor(labdata_agg$FY16cat,
      levels=c("[500,2e+05)","[250,500)","[100,250)","[1,100)","[0,1)"))

ggp <- ggplot(labdata_agg, aes(x=as.factor(AttendenceEvent),y=percent))

ggp + geom_bar(stat="identity", aes(fill=FY16cat))+
  ylab("% of alumni in each category")+xlab("Donation category for 2015")+
  scale_fill_discrete(name="Donation Category for 2016")+
  xlab("Attendance Event")+scale_x_discrete(labels=c("Didn't Attend", "Attended"))
```



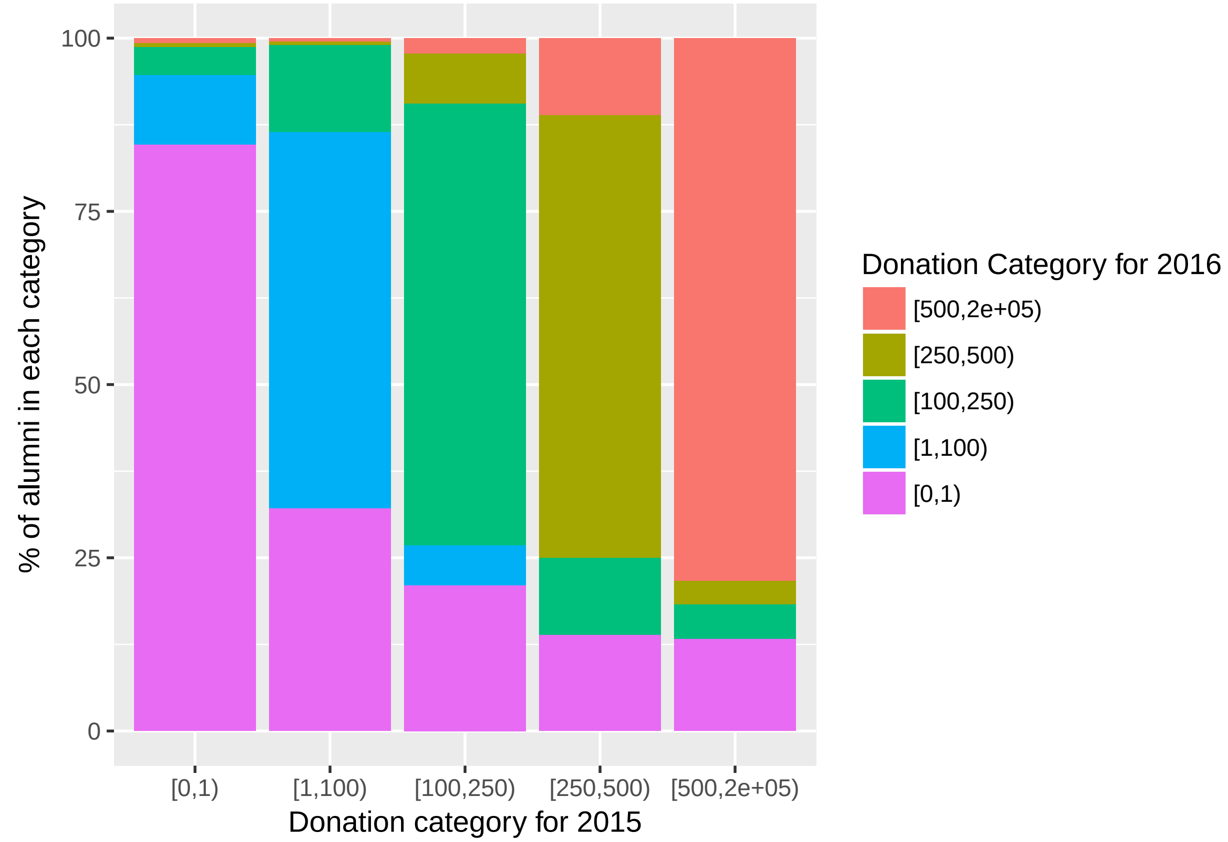
Those who went to alumni events were much more likely to donate and especially more likely to donate in the higher categories.

Donation category for the previous year (2015)

```
labdata_counts <- with(labdata,
                       aggregate(FY15cat,
                                 list(FY15cat=FY15cat),
                                 length))
labdata_agg <- with(labdata,
                    aggregate(FY15cat, list(FY15cat=FY15cat,
                                            FY16cat=FY16cat),
                    length))
labdata_agg <- merge(labdata_agg, labdata_counts, by="FY15cat")
labdata_agg <- setNames(labdata_agg, c("FY15cat", "FY16cat", "Count", "TotalCount"))
labdata_agg$percent <- 100*labdata_agg$Count/labdata_agg$TotalCount
labdata_agg$FY16cat <- factor(labdata_agg$FY16cat,
      levels=c("[500,2e+05)","[250,500)","[100,250)","[1,100)","[0,1)"))

ggp <- ggplot(labdata_agg, aes(x=FY15cat,y=percent))

ggp + geom_bar(stat="identity", aes(fill=FY16cat))+
  ylab("% of alumni in each category")+xlab("Donation category for 2015")+
  scale_fill_discrete(name="Donation Category for 2016")
```



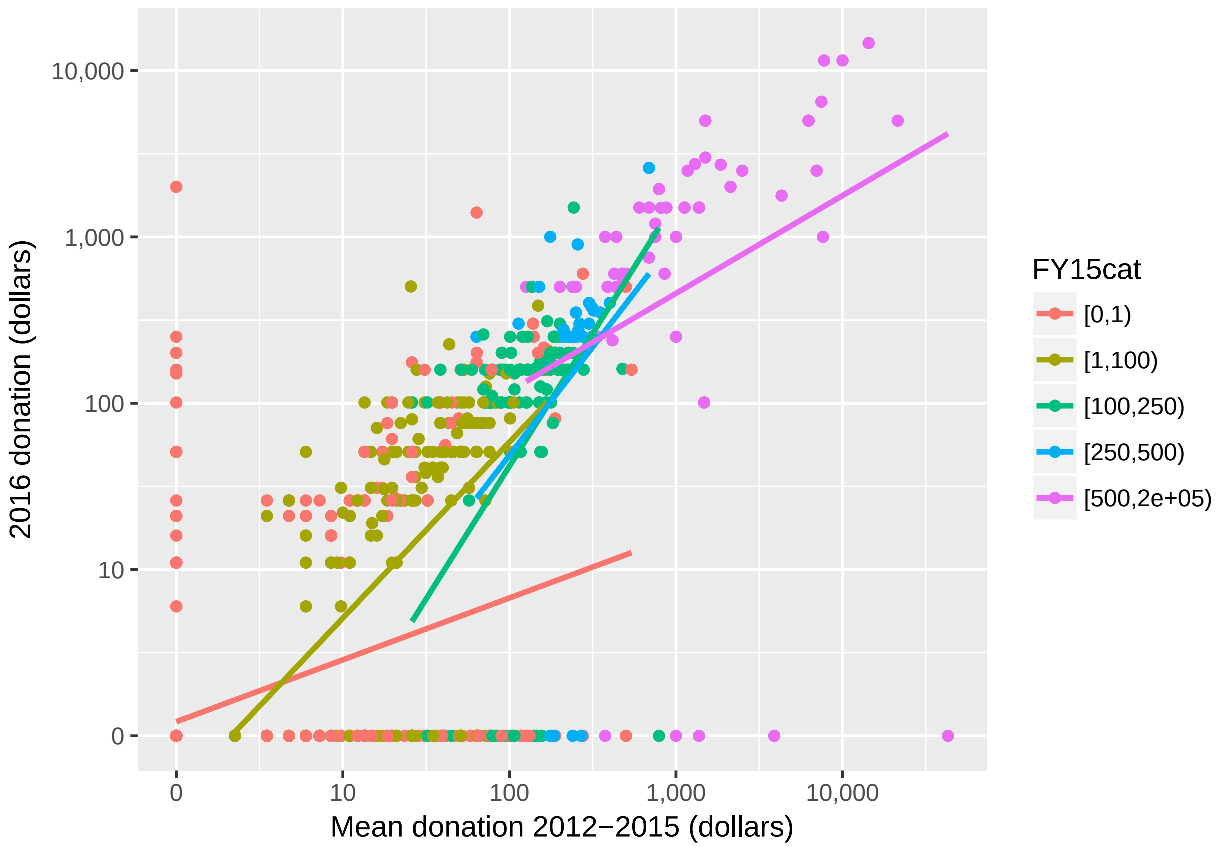
This stacked bar graph shows what proportion of alumni that fit into donation category X went into the same - or different - category in 2016. As expected, the largest bar in each group represents the same category. That is, the majority of those who donated $0 in 2015 also donated $0 in 2016, the majority of those who donated between $1-$100 in 2015 stayed in that category the next year, etc.

A key takeaway from this visualization is the relative instability of the [1,100) class - compared to other classes, this group was the least likely to donate in the same category. Still, past donations seem to be important in predicting future donations.

EDA - Interaction between last year's donations and overall donations in predicting 2016 donations

```
ggp <- ggplot(labdata, aes(x=log(meandonation+1)/log(10), y=log(FY16Giving+1)/log(10),
                           group=FY15cat, color=FY15cat))

ggp + geom_point() + geom_smooth(method="lm", se=F)+
  xlab("Mean donation 2012-2015 (dollars)")+ylab("2016 donation (dollars)")+
  scale_x_continuous(breaks=c(0,1,2,3,4), labels=c("0", "10", "100", "1,000", "10,000"))+
    scale_y_continuous(breaks=c(0,1,2,3,4), labels=c("0", "10", "100", "1,000", "10,000"))
```



Above is a scatterplot with the mean donation from 2012-2015 on the x-axis and the amount donated in 2016 on the y-axis. (While we are analyzing 2016 donations in categories, we thought this visualization was best done with the dollar amount). Overall, there is a clear relationship between mean donation and amount donated in 2016 - alumni typically didn't donate a drastically different amount in 2016 compared to how they've donated in years past. Of course, the exceptions are the many alumni who didn't donate in 2016 despite donating in years past (the dots on the horizontal x=0 line). There are a lot less alumni who donated in 2016 for the first time (the dots on the vertical y=0 line)

The purpose of the different colors/trend lines is to examine an interaction effect we found interesting and potentially useful for our model. The lines seem to generally be parallel except for the one representing the [0,1) category. The implication of this is that for alumni who donated in 2015, it is easier to predict their 2016 donations from their previous years, but for alumni who did not donate in 2015 the relationship is less clear. Rather than modeling an interaction term for each category for 2015, which could get too complex, it seems the interaction comes from whether they donated at all last year or not, which is why we will model their 2015 donations as a binary variable.

Statistical Modelling

Modeling

We examined both multinomial and ordinal regression models. In the following sections we explain our modelling steps, the model progression we went through and final model selection.

Explanation vs Prediction

As we mentioned before, the overall goal of the current work is prediction of future donations rather than explanation of the factors that drive donation. Given that, we choose to choose models based on their predictive power rather than their explanatory power.

For the same reason, we take a machine learning approach to model welection, where We split the data into a test and training set allowing us to test the accuracy of the model.

```
library(car)
set.seed(107)
sample <- sample.int(n = nrow(labdata), size = floor(.75*nrow(labdata)), replace = F)
train <- labdata[sample, ]
test  <- labdata[-sample, ]
```

Base Multinomial Model

Even though our dependent variable is ordinal, it has a categorical nature and sometimes multinomial models can excel at modelling this data.

Initially we started with a model with most of the original predictors from the data included in our EDA.

```
mn.model <- multinom(FY16cat ~ AttendenceEvent + Gender + Class.Year  + Marital.Status + NextDegreeBina
```

```
## # weights:  65 (48 variable)
## initial  value 1207.078434
## iter  10 value 823.384481
## iter  20 value 819.992249
## iter  30 value 724.396712
## iter  40 value 698.124498
## iter  50 value 691.095259
## iter  60 value 689.640743
## iter  70 value 688.288348
## iter  80 value 688.083970
## iter  90 value 687.895492
## final  value 687.728522
## converged
```

```
mn.model$AIC
```

```
## [1] 1471.457
```

We can now do hypothesis testing, in this case through analysis of variance.

```
Anova(mn.model)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: FY16cat
##                 LR Chisq Df Pr(>Chisq)
## AttendenceEvent   25.5840  4  3.838e-05 ***
## Gender             7.2793  4  0.1218443
## Class.Year         8.8743  4  0.0643199 .
## Marital.Status    27.5872 12  0.0063545 **
## NextDegreeBinary  25.3854  4  4.208e-05 ***
## FY12Giving        21.3515  4  0.0002697 ***
## FY13Giving        23.3371  4  0.0001084 ***
## FY14Giving         6.3241  4  0.1762190
## FY15Giving        31.0752  4  2.955e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Using analysis of variance test we obtain the analysis of deviance table, where we can see that for all variables except for FY14Giving and Gender, we can reject the null hypothesis that the β parameter for those predictors is 0.

To understand strengths and weaknesses of our base model, we use a confusion matrix, which provides rich information about what categories we fail to predict, where we do well, etc.

```r
mn.preds <- predict(mn.model, test, type="class")
confusionMatrix(mn.preds, test$FY16cat)
```

```
## Confusion Matrix and Statistics
##
##               Reference
## Prediction    [0,1) [1,100) [100,250) [250,500) [500,2e+05)
##   [0,1)         133      48        34         4           1
##   [1,100)         0       0         0         0           0
##   [100,250)       2       0         5         8           3
##   [250,500)       0       1         1         1           1
##   [500,2e+05)     0       0         0         0           8
##
## Overall Statistics
##
##                Accuracy : 0.588
##                  95% CI : (0.5242, 0.6496)
##     No Information Rate : 0.54
##     P-Value [Acc > NIR] : 0.0719
##
##                   Kappa : 0.1934
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                       Class: [0,1) Class: [1,100) Class: [100,250)
## Sensitivity                 0.9852          0.000           0.1250
## Specificity                 0.2435          1.000           0.9381
## Pos Pred Value              0.6045            NaN           0.2778
## Neg Pred Value              0.9333          0.804           0.8491
## Prevalence                  0.5400          0.196           0.1600
## Detection Rate              0.5320          0.000           0.0200
## Detection Prevalence        0.8800          0.000           0.0720
## Balanced Accuracy           0.6143          0.500           0.5315
##                       Class: [250,500) Class: [500,2e+05)
## Sensitivity                    0.07692             0.6154
## Specificity                    0.98734             1.0000
## Pos Pred Value                 0.25000             1.0000
## Neg Pred Value                 0.95122             0.9793
## Prevalence                     0.05200             0.0520
## Detection Rate                 0.00400             0.0320
## Detection Prevalence           0.01600             0.0320
## Balanced Accuracy              0.53213             0.8077
```

The confusion matrix shows that the model is better at predicting the [0,1) and the [500, 2e+05) groups and has trouble with the middle groups, specifically the [250,500). This group has a small sample size in our dataset which would make this group harder to predict.

Multinomial Model

We now aim to improve our model using engineers features from our EDA. Interestingly, features that we believed would increase our predictive power, actually had a very positive impact in the overall performance of our model.

```
mn.model <- multinom(FY16cat ~ AttendenceEvent + Gender + NextDegreeBinary + as.ordered(YearsSinceGrad)
```

```
## # weights:  60 (44 variable)
## initial  value 1207.078434
## iter  10 value 673.856603
## iter  20 value 549.968675
## iter  30 value 541.035594
## iter  40 value 540.351403
## iter  50 value 540.163299
## iter  60 value 540.091376
## final  value 540.089293
## converged
```

```
summary(mn.model)
```

```
## Call:
## multinom(formula = FY16cat ~ AttendenceEvent + Gender + NextDegreeBinary +
##     as.ordered(YearsSinceGrad) + log(meandonation + 1) * gaveLastYear,
##     data = train)
##
## Coefficients:
##             (Intercept) AttendenceEvent    GenderM NextDegreeBinary
## [1,100)       -3.201725      -0.2001595 -0.4253529        0.7018319
## [100,250)     -5.487293       0.3577857  0.4480592        0.2963986
## [250,500)     -6.386507       1.0273326  0.1851387        0.1591967
## [500,2e+05)   -8.674420       0.9700352  0.0362706        1.3196900
##             as.ordered(YearsSinceGrad).L as.ordered(YearsSinceGrad).Q
## [1,100)                      -0.68213895                  -0.07795757
## [100,250)                    -0.30003898                   0.31422705
## [250,500)                     0.01523424                  -0.01343015
## [500,2e+05)                  -0.48866063                   0.33303881
##             as.ordered(YearsSinceGrad).C as.ordered(YearsSinceGrad)^4
## [1,100)                       -0.1930499                  -0.05416405
## [100,250)                     -0.3834212                  -0.06846867
## [250,500)                     -0.4972663                   0.17389623
## [500,2e+05)                    0.5429951                  -0.27048164
##             log(meandonation + 1) gaveLastYear
## [1,100)                 0.4083159     3.684256
## [100,250)               0.7973904    -0.558694
## [250,500)              -8.3668849    -4.451155
## [500,2e+05)             1.0020829    -9.658138
##             log(meandonation + 1):gaveLastYear
## [1,100)                             -0.5933065
## [100,250)                            0.4809666
## [250,500)                           10.2785845
## [500,2e+05)                          2.1361589
##
## Std. Errors:
##             (Intercept) AttendenceEvent   GenderM NextDegreeBinary
## [1,100)       0.3407006       0.2595666 0.2449338        0.2667601
## [100,250)     0.6488573       0.3275329 0.2939130        0.3198344
## [250,500)     1.1639328       0.6300943 0.4828732        0.5462899
## [500,2e+05)   1.6713528       0.6541512 0.4930611        0.6556388
##             as.ordered(YearsSinceGrad).L as.ordered(YearsSinceGrad).Q
## [1,100)                        0.3593229                    0.3313874
## [100,250)                      0.3737672                    0.3466673
## [250,500)                      0.6860883                    0.6120627
## [500,2e+05)                    0.7251682                    0.6455318
##             as.ordered(YearsSinceGrad).C as.ordered(YearsSinceGrad)^4
## [1,100)                        0.2993100                    0.2792084
## [100,250)                      0.3290007                    0.3156834
## [250,500)                      0.5673585                    0.4986111
## [500,2e+05)                    0.5656347                    0.5215279
##             log(meandonation + 1) gaveLastYear
## [1,100)                 0.1006932    0.6741158
## [100,250)               0.1560895    1.0477822
## [250,500)               0.1634199    1.8710405
## [500,2e+05)             0.3531793    2.4170102
##             log(meandonation + 1):gaveLastYear
## [1,100)                              0.1949353
## [100,250)                            0.2574907
## [250,500)                            0.1634076
## [500,2e+05)                          0.5035547
##
## Residual Deviance: 1080.179
## AIC: 1168.179
```

f We can see that were able to reduce the model AIC to 1168 and reduce the features. Reducing the number of features should help with not overfitting the model on a relatively small sample set. We removed marital status because of small sample sizes for the Widow and Divorced groups. The addition of $log(meandonation + 1)$ and $gaveLastYear$ provide powerful and yet concise insights on the past donation history.

```r
Anova(mn.model)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: FY16cat
##                                    LR Chisq Df Pr(>Chisq)
## AttendenceEvent                       5.937  4    0.20389
## Gender                                8.140  4    0.08660 .
## NextDegreeBinary                     10.852  4    0.02828 *
## as.ordered(YearsSinceGrad)           12.106 16    0.73667
## log(meandonation + 1)               204.803  4  < 2.2e-16 ***
## gaveLastYear                         52.719  4  9.758e-11 ***
## log(meandonation + 1):gaveLastYear   40.848  4  2.890e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Analyzing the deviance table, we can boserve that attendence to recruiting events and years since graduation do not pass the likelihood test, accepting the null hypothesis that the coefficient for those predictors is zero. However, from a domain knowledge perspective we feel that those variables are important, and that is confirmed by the improved prediction power of our model, which we can analyze by showing its confusion matrix.

```r
mn.preds <- predict(mn.model, test, type="class")
confusionMatrix(mn.preds, test$FY16cat)
```

```
## Confusion Matrix and Statistics
##
##               Reference
## Prediction     [0,1) [1,100) [100,250) [250,500) [500,2e+05)
##   [0,1)          117      25        14         1           0
##   [1,100)         12      22         5         0           0
##   [100,250)        4       2        21        12           4
##   [250,500)        0       0         0         0           0
##   [500,2e+05)      2       0         0         0           9
##
## Overall Statistics
##
##                Accuracy : 0.676
##                  95% CI : (0.6142, 0.7336)
##     No Information Rate : 0.54
##     P-Value [Acc > NIR] : 8.29e-06
##
##                   Kappa : 0.4604
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: [0,1) Class: [1,100) Class: [100,250)
## Sensitivity                0.8667         0.4490           0.5250
## Specificity                0.6522         0.9154           0.8952
## Pos Pred Value             0.7452         0.5641           0.4884
## Neg Pred Value             0.8065         0.8720           0.9082
## Prevalence                 0.5400         0.1960           0.1600
## Detection Rate             0.4680         0.0880           0.0840
## Detection Prevalence       0.6280         0.1560           0.1720
## Balanced Accuracy          0.7594         0.6822           0.7101
##                      Class: [250,500) Class: [500,2e+05)
## Sensitivity                     0.000             0.6923
## Specificity                     1.000             0.9916
## Pos Pred Value                    NaN             0.8182
## Neg Pred Value                  0.948             0.9833
## Prevalence                      0.052             0.0520
## Detection Rate                  0.000             0.0360
## Detection Prevalence            0.000             0.0440
## Balanced Accuracy               0.500             0.8419
```

The confusion matrix shows a greatly improved overall acuracy with at 67.6% with a 95% confidence between 61.4% and 73.3%. We also see an improvement in the sensitivity for the highest donor group, which to us is one of the most important groups to get correctly predicted. It is important to be able to identify individuals in this group since they account for ~77% of donations in 2016.

Ordinal Model

Our final model was an oridinal model, using the same predictors as our improved multinomial model. The ordinal model improved slightly on overall accuracy with our test set scoring a 68.8%. The ordinal model showed improvements particularly the sensitivity and specificity scores for the highest donor and the non-donor groups.

Final Model Selection: Ordinal model

To select and compare models, we used statistical analysis such as AIC and hypothesis testing, but also we took a holistic approach by analyzing confusion matrices for each model and understanding and interpreting each model's strengths and weaknesses. Particularly, we felt that detecting individuals that donated in the highest amount group and individuals that don't donate at all was very valuable, while differentiating between the two middle donation brackets was not so important. Not only that, but ~70% of donations during 2016 were in the highest bracket and the non-donor group was the majority of our sample, so even a better reason to prioritize the correct prediction of those group. Confusion matrices allow us to compare how each model did on each group, which is valuable given our approach.

```
ord.model <- clm(FY16cat ~ AttendenceEvent + Gender + NextDegreeBinary + as.ordered(YearsSinceGrad) + l
summary(ord.model)
```

```
## formula:
## FY16cat ~ AttendenceEvent + Gender + NextDegreeBinary + as.ordered(YearsSinceGrad) + log(meandonatio
## data:    train
##
##  link  threshold nobs logLik  AIC     niter max.grad cond.H
##  logit flexible  750  -574.26 1176.51 6(0)  8.19e-13 2.0e+03
##
## Coefficients:
##                                     Estimate Std. Error z value Pr(>|z|)
## AttendenceEvent                      0.13502    0.19723   0.685  0.49359
## GenderM                              0.07007    0.17549   0.399  0.68970
## NextDegreeBinary                     0.42749    0.19634   2.177  0.02946
## as.ordered(YearsSinceGrad).L        -0.39603    0.22815  -1.736  0.08259
## as.ordered(YearsSinceGrad).Q         0.47129    0.21071   2.237  0.02531
## as.ordered(YearsSinceGrad).C        -0.32409    0.20592  -1.574  0.11552
## as.ordered(YearsSinceGrad)^4         0.05643    0.19958   0.283  0.77736
## log(meandonation + 1)                0.62382    0.08652   7.211 5.57e-13
## gaveLastYear                        -1.26310    0.48943  -2.581  0.00986
## log(meandonation + 1):gaveLastYear   0.72544    0.13639   5.319 1.04e-07
##
## AttendenceEvent
## GenderM
## NextDegreeBinary                    *
## as.ordered(YearsSinceGrad).L        .
## as.ordered(YearsSinceGrad).Q        *
## as.ordered(YearsSinceGrad).C
## as.ordered(YearsSinceGrad)^4
## log(meandonation + 1)               ***
## gaveLastYear                        **
## log(meandonation + 1):gaveLastYear ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Threshold coefficients:
##                         Estimate Std. Error z value
## [0,1)|[1,100)             3.3214     0.2970   11.18
## [1,100)|[100,250)         4.9091     0.3299   14.88
## [100,250)|[250,500)       6.9660     0.3827   18.20
## [250,500)|[500,2e+05)     7.8569     0.4147   18.95
```

Note that the ordinal model also maintains the parsimony of the previous model, using the same predictors. We can observe that most variables are statistically significant, except for attendenceEven, Gender, and some levels of the years since graduation. However, these variables proved to be quite valuable in our EDA, and furthermore we did formal hypothesis testing of removing them, and the model with them proved to be better, and also had better predictive power which again is our focus in this study.

```r
Anova(ord.model)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: FY16cat
##                                    Df    Chisq Pr(>Chisq)
## AttendenceEvent                     1 125.0837    < 2e-16 ***
## Gender                              1 221.4749    < 2e-16 ***
## NextDegreeBinary                    1 331.3069    < 2e-16 ***
## as.ordered(YearsSinceGrad)          4 397.0191    < 2e-16 ***
## log(meandonation + 1)               1   2.6241    0.10525
## gaveLastYear                        1   6.5096    0.01073 *
## log(meandonation + 1):gaveLastYear  1   2.4771    0.11552
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can now analyze the analysis of deviance table above for the ordinal model, where the null hypothesis is that the β coefficient for each predictor is 0 and the alternate hypothesis is that $\beta \neq 0$. Note that we fail to reject the null hypothesis for the $log(meandonation + 1)$ and for the interaction term. However these terms proved to have great predictive power, and greatly increase the parsimony of our model.

```r
ord.preds <- predict(ord.model, test, type="class")
confusionMatrix(ord.preds$fit, test$FY16cat)
```

```
## Confusion Matrix and Statistics
##
##               Reference
## Prediction    [0,1) [1,100) [100,250) [250,500) [500,2e+05)
##   [0,1)         122      30         8         1           0
##   [1,100)         8      16         8         1           0
##   [100,250)       4       3        24        10           3
##   [250,500)       0       0         0         0           0
##   [500,2e+05)     1       0         0         1          10
##
## Overall Statistics
##
##                Accuracy : 0.688
##                  95% CI : (0.6266, 0.7449)
##     No Information Rate : 0.54
##     P-Value [Acc > NIR] : 1.292e-06
##
##                   Kappa : 0.4763
##  Mcnemar's Test P-Value : NA
```

```
##
## Statistics by Class:
##
##                      Class: [0,1) Class: [1,100) Class: [100,250)
## Sensitivity                0.9037         0.3265           0.6000
## Specificity                0.6609         0.9154           0.9048
## Pos Pred Value             0.7578         0.4848           0.5455
## Neg Pred Value             0.8539         0.8479           0.9223
## Prevalence                 0.5400         0.1960           0.1600
## Detection Rate             0.4880         0.0640           0.0960
## Detection Prevalence       0.6440         0.1320           0.1760
## Balanced Accuracy          0.7823         0.6210           0.7524
##                      Class: [250,500) Class: [500,2e+05)
## Sensitivity                     0.000             0.7692
## Specificity                     1.000             0.9916
## Pos Pred Value                    NaN             0.8333
## Neg Pred Value                  0.948             0.9874
## Prevalence                      0.052             0.0520
## Detection Rate                  0.000             0.0400
## Detection Prevalence            0.000             0.0480
## Balanced Accuracy               0.500             0.8804
```

We can observe that this confusion matrix yields not only the best accuracy of all our models, but also the best predictions for the highest and lowest donation bracket, which is aligned with our goals for this study.

Another interesting thing about the ordinal model presented is that the failures to predict tend to be in categories that are close in terms of domain meaning, for example, a mispredicted high donator may really be a moderate donator, but errors tend to group around the nearby categories of the expected one. On the contrary in the multinomial model we observe slightly more deviation in the errors, which makes sense from an interpretative point of view since in the ordinal model we are bringing in information about the order of the output categories.

Final Remarks

In our analysis we have shown the relationships between several independent variables and the 2016 donations of alumni, and shown our model to have good predictive power for several of the categories. Overall, our model is able to predict donors vs. non-donors with high accuracy. Both previous donation patterns, as well as other factors like attending alumni events and having an advanced degree, allow us to discriminate between those who donated nothing in 2016 and those who donated something.

An additional goal of the model was to be able to predict which category of donation an alumnus would fall under, which would allow fundraising campaigns to not only target alumni who will donate but also pick out those who are likely to donate the most. In that regard our model fares more poorly. We are able to predict who will fall into the highest bucket of donation [500,2e+05) fairly well, as alumni who donate that much tend to do it on a yearly basis. However, our model struggled with accurately predicting the other three categories. A big reason for this is the sample size. Out of 1000 alumni, only 39 donated in the [250,500) category, which contributes to the lack of accuracy in predicting that particular category.

The administration can gain insight from our model on alumni who will donate vs. those who won't. It is clear that previous donation patterns have an influence on future donation paterns, and those should be taken as the most important predictive factors. It would be particularly useful to focus efforts on alumni who have donated the year before. But the administration can also use the information from our model to target those who attend alumni events and those with an advanced degree. The trend of older alumni donating more is also interesting. Presumably older alumni have larger incomes and therfore are able to donate more, and would be a good group to focus efforts on.

There is a lot of information that, if made available, would improve the quality of our models significantly. Many of the trends we noticed appear to have to do with income - those with an advanced degree, as well as older alumni, probably have larger incomes and can donate more. If the administration was able to collect information about income - or, more realistically, employment status of some sort - that information could be very useful. Another piece of information that could help would be involvement of the alumni while they were in college. One would expect that alumni that were more involved in extracurriculars - sports, theater, student government, etc. - would feel a greater connection to their university and therefore be more likely to donate. Race would also be an interesting variable to investigate; while we don't have an a priori hypothesis for its effect on donation, if the information was available it could potentially be useful. Finally, it would be useful to have information about their family's relationship to the university. If an alumnus was married to another alumnus, or has children who attend the university, that would likely increase the chances of them donating.

One final note is that the buckets chosen for the FY16Giving do not fit the data well. Unless the buckets represent some well-established standard, we believe that better buckets could be constructed to represent the distribution of donations more evenly. As mentioned before, a big challenge for the model was how sparsely represented some of the categories were, and slicing the data in a different way could improve our models substantially.