W241_Project_PGSS_Campaign

Subha Vadakkumkoor, Walter Burge

August 3, 2018

```
knitr::opts_chunk$set(echo = TRUE)
```

```
library(data.table)
library(stargazer)
```

```
##
## Please cite as:
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:data.table':
##
##     between, first, last
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readr)
```

Loading data

Data imported from Salsa, excluding PII fields (name and email address) are read into R dataframes. All dataframes have the same structure and format. With each step of the treatment (Orignal email, Reminder1 and Reminder2), there are 2 files: list of people that were assigned the treatment (recieved the email) and list of people that responded to the treatment (donated money).

```
#Load data

#Original email
orig_email_rec<-read.csv('./data/BlastReport_Class Experiment Final Email_Recipients.csv')
setnames(orig_email_rec, old=c("Opened"), new=c("opened_orig_email"))

orig_email_resp<-read.csv('./data/BlastReport_Class Experiment Final Email_Conversions.csv')
sapply(orig_email_resp, class)
```

```
##    Supporter.ID     External.ID         Country           State
##        "factor"       "integer"        "factor"        "factor"
##            City  Reference.Name      Split.Name       Time.Sent
##        "factor"        "factor"        "factor"        "factor"
```

```
## Conversion.Date   Activity.Type   Activity.Name     Activity.ID
##        "factor"        "factor"        "factor"        "factor"
##          Amount   Donation.Type
##       "numeric"        "factor"
```

```
#Create an indicator and rename columns to reflect original email response (useful for merge later)
orig_email_resp$donated_after_orig_email=1
names(orig_email_resp)
```

```
##  [1] "Supporter.ID"             "External.ID"
##  [3] "Country"                  "State"
##  [5] "City"                     "Reference.Name"
##  [7] "Split.Name"               "Time.Sent"
##  [9] "Conversion.Date"          "Activity.Type"
## [11] "Activity.Name"            "Activity.ID"
## [13] "Amount"                   "Donation.Type"
## [15] "donated_after_orig_email"
```

```
setnames(orig_email_resp, old=c("Conversion.Date","Amount"), new=c("Orig_email_conversion_date", "orig_

#Reminder1
reminder1_rec<-read.csv('./data/BlastReport_Class experiment Reminder1_Recipients.csv')
setnames(reminder1_rec, old=c("Opened"), new=c("opened_reminder1"))
reminder1_rec_subset=reminder1_rec[,c("opened_reminder1","Supporter.ID")]

reminder1_resp<-read.csv('./data/BlastReport_Class experiment Reminder1_Conversions.csv')
reminder1_resp$donated_after_reminder1=1
setnames(reminder1_resp, old=c("Conversion.Date","Amount"), new=c("reminder1_conversion_date", "reminde

#Reminder2
reminder2_rec<-read.csv('./data/BlastReport_Class Experiment Reminder 2_Recipients.csv')
setnames(reminder2_rec, old=c("Opened"), new=c("opened_reminder2"))
reminder2_rec_subset=reminder2_rec[,c("opened_reminder2","Supporter.ID")]


reminder2_resp<-read.csv('./data/BlastReport_Class Experiment Reminder 2_Conversions.csv')
reminder2_resp$donated_after_reminder2=1
setnames(reminder2_resp, old=c("Conversion.Date","Amount"), new=c("reminder2_conversion_date", "reminde

#Load donor profile file
alumni_profile<-read.csv('./data/Alumni_profile_all.csv',colClasses = c("character","character","numeri
sapply(alumni_profile,class)
```

```
##         Constituent.Number           Constituent.UUID
##                "character"                "character"
##                  PGSS.Year               Faculty.Year
##                  "numeric"                "character"
##                TA.RD.Years              Received.Date
##                "character"                "character"
##                 Gift.Count         Total.Gift.Amounts
##                "character"                "character"
##        Largest.Gift.Amount             Last.Gift.Date
##                "character"                "character"
##      Last.Gift.Amount.Ever Months_since_last_donation
##                "character"                  "numeric"
```

```
##           last_gift_amount          total_gift_amount
##                  "numeric"                  "numeric"
##                gift_count           LYBUNT_indicator
##                 "numeric"                  "numeric"
##           SYBUNT_Indicator              Never_donator
##                  "numeric"                  "numeric"
```

```
#names(alumni_profile)

#Examine the layout of a representative file
cat("Fields in recipients file\n")
```

```
## Fields in recipients file
```

```
#names(orig_email_rec)

cat("\nFields in responder files\n")
```

```
##
## Fields in responder files
```

```
#names(orig_email_resp)

#Get dimensions of each file
cat("\nDimensions of each file\n")
```

```
##
## Dimensions of each file
```

```
dfList <- list(orig_email_rec,orig_email_resp,reminder1_rec,reminder1_resp,reminder2_rec,reminder2_resp
lapply(dfList,dim)
```

```
## [[1]]
## [1] 2110   17
##
## [[2]]
## [1] 25 15
##
## [[3]]
## [1] 2107   17
##
## [[4]]
## [1] 37 15
##
## [[5]]
## [1] 2111   17
##
## [[6]]
## [1] 34 15
```

Now we merge the orignal rec and resp datasets with the responders from reminder1 and reminder2 and alumni profile.

```
#Merge original recipients email with Alumni profile
merged<-merge(orig_email_rec,alumni_profile,by.x="Supporter.ID",by.y="Constituent.UUID",all.x=TRUE)

#Merge with the original email response
merged<-merge(merged,orig_email_resp[,c("Supporter.ID","Orig_email_conversion_date","orig_email_amount"
```

```r
cat("\nNum of rows",nrow(merged))
```

```
##
## Num of rows 2110
```

```r
#Merge with the first reminder response
merged<-merge(merged,reminder1_rec_subset,by="Supporter.ID",all.x=TRUE)

merged<-merge(merged,reminder1_resp[,c("Supporter.ID","reminder1_conversion_date","reminder1_amount","d

cat("\nNum of rows",nrow(merged))
```

```
##
## Num of rows 2110
```

```r
#Merge with the second reminder response
merged<-merge(merged,reminder2_rec_subset,by="Supporter.ID",all.x=TRUE)

merged<-merge(merged,reminder2_resp[,c("Supporter.ID","reminder2_conversion_date","reminder2_amount","d

cat("\nNum of rows",nrow(merged))
```

```
##
## Num of rows 2110
```

```r
#Set NA's in indicators to 0
merged[(is.na(merged$donated_after_orig_email)),]$donated_after_orig_email=0
merged[(is.na(merged$donated_after_reminder1)),]$donated_after_reminder1=0
merged[(is.na(merged$donated_after_reminder2)),]$donated_after_reminder2=0

#Adding indicator for donated this year before treatment
merged$donated_TY_pre_treatment <- as.Date("1900-01-01")
merged[!is.na(merged$Last.Gift.Date),]$donated_TY_pre_treatment <- as.Date(merged[!is.na(merged$Last.Gi
#Changed this to exclude donations after the original email was sent, assuming this is for 2a. So now n
merged$donated_TY_pre_treatment <- ifelse(merged$donated_TY_pre_treatment>as.Date("2017-12-31") & merge
print("Number of donations this year pre-treatment: ")
```

```
## [1] "Number of donations this year pre-treatment: "
```

```r
print(sum(merged$donated_TY_pre_treatment))
```

```
## [1] 0
```

Let us create dependant and covariates

```r
#Check for MULTIPLE DONATIONS

#Define treatment indicator
merged$treat<-ifelse(merged$Split.Name %in% c("Split A"),1,0)
table(merged$Split.Name,merged$treat)
```

```
##
##              0    1
##   Split A    0 1055
##   Split B 1055    0
```

```r
#Total donation amount
merged$orig_email_amount_copy=merged$orig_email_amount
```

```
merged$reminder1_amount_copy=merged$reminder1_amount
merged$reminder2_amount_copy=merged$reminder2_amount

merged$orig_email_amount=ifelse(is.na(merged$orig_email_amount_copy),0,merged$orig_email_amount)
merged$reminder1_amount=ifelse(is.na(merged$reminder1_amount_copy),0,merged$reminder1_amount_copy)
merged$reminder2_amount=ifelse(is.na(merged$reminder2_amount_copy),0,merged$reminder2_amount_copy)

merged$total_donation_amount=merged$orig_email_amount+merged$reminder1_amount+merged$reminder2_amount
summary(merged$total_donation_amount)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    0.000    0.000    0.000    5.583    0.000 4000.000
```

```
summary(merged$orig_email_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   1.121   0.000 500.000
```

```
summary(merged$reminder1_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   1.062   0.000 500.000
```

```
summary(merged$reminder2_amount)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0     3.4     0.0  4000.0
```

```
#Days till donation
merged$donation_date=coalesce(as.Date(merged$Orig_email_conversion_date),as.Date(merged$reminder1_conve
#merged[(is.na(merged$donation_date)),]$donation_date=as.Date('2018-7-24')
merged$days_till_donation=merged$donation_date-as.Date(merged$Time.Sent)
#merged[(merged$donation_date>0),]
table(merged$days_till_donation)
```

```
##
##  0  1  2  3  4  5  6  7  8  9
##  6 11  1  1 26  8  2 32  3  1
```

```
#Donation response indicator
merged$donated_any_time=0
merged[!is.na(merged$days_till_donation),]$donated_any_time=1

#OPened any reminder
merged$opened_atleast_one_reminder=merged$opened_reminder1 * merged$opened_reminder2
merged$opened_any_email=merged$opened_orig_email * merged$opened_atleast_one_reminder

#PGSS year based
merged$Ben_involved_ind=ifelse(merged$PGSS.Year>=1997,1,0)
merged$Ben_involved_ind[merged$PGSS.Year==1998]=0
#table(merged$Ben_involved_ind,merged$PGSS.Year)

#Batch age indicator
merged$PGSS_age=merged$PGSS.Year-1982

#Older batch indicator
merged$PGSS_age_bucket=merged$PGSS_age%/%5
```

```
merged$PGSS_age_bucket[merged$PGSS.Year>=2007]=6
#table(merged$PGSS.Year,merged$PGSS_age_bucket)


#Interaction of split and Ben involvement
merged$treat_Ben_inter=merged$treat * merged$Ben_involved_ind
merged$treat_reminder=merged$treat * merged$opened_reminder1 * merged$opened_reminder2

#What else do we need interaction or difference in difference for
```

```
#Define non-compliance. What about people who opened but did not click or contribute?

#Check for emails that were read within the experiment time period: DO we need to stop looking for dona
merged1<-merged[(merged$opened_reminder2>as.Date("2017-08-02") ),]
nrow(merged1)
```

```
## [1] 11
```

```
#Method1: Exclude anyone who did not open original email (message is the treatment)
merged$compliant_orig=0
merged[merged$opened_orig_email,]$compliant_orig=1
compliant_orig<-merged[merged$compliant_orig==1,]

#Method2: Subject line is the treatment so everyone is treated
merged$compliant_all=1

#Check number of people who typically do not open emails in previous campaigns
```

Stats

```
cat("Response rate after original email")
```

```
## Response rate after original email
```

```
table(merged$donated_after_orig_email,merged$treat)
```

```
##
##        0    1
##   0 1048 1038
##   1    7   17
```

```
cat("Response rate after reminder1")
```

```
## Response rate after reminder1
```

```
table(merged$donated_after_reminder1,merged$treat)
```

```
##
##        0    1
##   0 1039 1036
##   1   16   19
```

```
cat("Response rate after reminder2")
```

```
## Response rate after reminder2
```

```
table(merged$donated_after_reminder2,merged$treat)
```

```
##
```

```
##        0    1
##   0 1039 1039
##   1   16   16
```

```
table(merged$opened_orig_email,merged$donated_after_reminder1)
```

```
##
##            0    1
##   FALSE 1243    7
##   TRUE   832   28
```

```
table(merged$opened_orig_email,merged$donated_after_reminder2)
```

```
##
##            0    1
##   FALSE 1240   10
##   TRUE   838   22
```

Checking for bad control: Does not seem like the reminders are bad control

```
#table(merged$opened_orig_email,merged$opened_reminder1)

#Regress reminder against original email

reg_remember_treatment<-lm(opened_reminder1~merged$Split.Name,data=merged)
print(summary(reg_remember_treatment))
```

```
##
## Call:
## lm(formula = opened_reminder1 ~ merged$Split.Name, data = merged)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4463 -0.4463 -0.4291  0.5537  0.5709
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)                0.42912    0.01531  28.032   <2e-16 ***
## merged$Split.NameSplit B   0.01723    0.02164   0.796    0.426
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4963 on 2102 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.0003015,  Adjusted R-squared:  -0.0001741
## F-statistic: 0.6339 on 1 and 2102 DF,  p-value: 0.426
```

```
reg_remember_treatment<-lm(opened_reminder2~merged$Split.Name,data=merged)
print(summary(reg_remember_treatment))
```

```
##
## Call:
## lm(formula = opened_reminder2 ~ merged$Split.Name, data = merged)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3060 -0.3060 -0.3038  0.6940  0.6962
```

```
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                0.306006   0.014221  21.518   <2e-16 ***
## merged$Split.NameSplit B -0.002196   0.020106  -0.109    0.913
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4606 on 2097 degrees of freedom
##   (11 observations deleted due to missingness)
## Multiple R-squared:  5.689e-06,  Adjusted R-squared:  -0.0004712
## F-statistic: 0.01193 on 1 and 2097 DF,  p-value: 0.913
```

```
reg_remember_treatment<-lm(opened_atleast_one_reminder~merged$Split.Name,data=merged)
print(summary(reg_remember_treatment))
```

```
##
## Call:
## lm(formula = opened_atleast_one_reminder ~ merged$Split.Name,
##     data = merged)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.240 -0.240 -0.225 -0.225  0.775
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 0.22498    0.01305  17.244   <2e-16 ***
## merged$Split.NameSplit B  0.01502    0.01845   0.814    0.415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4226 on 2097 degrees of freedom
##   (11 observations deleted due to missingness)
## Multiple R-squared:  0.0003162,  Adjusted R-squared:  -0.0001605
## F-statistic: 0.6634 on 1 and 2097 DF,  p-value: 0.4155
```

Regression for response

```
#names(merged)
```

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(sandwich)
#Function to return robust errors
printrobustSE <- function(regmodel)
  {
  regmodel$vcovHC <- vcovHC(regmodel)
  summ <- coeftest(regmodel, regmodel$vcovHC)
```

```
  return(summ)
}
```

Run

```
merged$donated_any_time <- factor(merged$donated_any_time)


#Method1: exclude everyone who did not open original email
reg_response<-glm(donated_any_time~treat+opened_atleast_one_reminder+PGSS_age+last_gift_amount+treat_Ben
print(summary(reg_response))
```

```
##
## Call:
## glm(formula = donated_any_time ~ treat + opened_atleast_one_reminder +
##     PGSS_age + last_gift_amount + treat_Ben_inter + SYBUNT_Indicator +
##     LYBUNT_indicator + gift_count, family = "binomial", data = compliant_orig)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9100  -0.4614  -0.2835  -0.1726   3.0359
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -5.1323881  0.5836513  -8.794  < 2e-16 ***
## treat                        0.7252654  0.3781138   1.918   0.0551 .
## opened_atleast_one_reminder  1.1204814  0.2798384   4.004 6.23e-05 ***
## PGSS_age                     0.0230962  0.0170851   1.352   0.1764
## last_gift_amount            -0.0001289  0.0004993  -0.258   0.7963
## treat_Ben_inter             -0.6068334  0.4490516  -1.351   0.1766
## SYBUNT_Indicator             1.0793421  0.5115865   2.110   0.0349 *
## LYBUNT_indicator             2.0010542  0.4212508   4.750 2.03e-06 ***
## gift_count                   0.0281849  0.0180428   1.562   0.1183
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 493.73  on 852  degrees of freedom
## Residual deviance: 431.15  on 844  degrees of freedom
##   (7 observations deleted due to missingness)
## AIC: 449.15
##
## Number of Fisher Scoring iterations: 6
```

```
#Methods: all
#THis is intent to treat
reg_response<-glm(donated_any_time~treat+opened_atleast_one_reminder+PGSS_age+last_gift_amount+treat_Ben
print(printrobustSE(reg_response))
```

```
##
## z test of coefficients:
```

```
##
##                               Estimate  Std. Error  z value  Pr(>|z|)
## (Intercept)                 -6.0659e+00  5.1809e-01 -11.7082 < 2.2e-16 ***
## treat                        6.9154e-01  3.5042e-01   1.9734  0.048446 *
## opened_atleast_one_reminder  1.6833e+00  2.5302e-01   6.6530 2.872e-11 ***
## PGSS_age                     3.0318e-02  1.7198e-02   1.7629  0.077922 .
## last_gift_amount            -3.3525e-05  2.5725e-04  -0.1303  0.896311
## treat_Ben_inter             -5.7107e-01  4.2977e-01  -1.3288  0.183918
## SYBUNT_Indicator             1.3001e+00  4.2883e-01   3.0318  0.002431 **
## LYBUNT_indicator             2.1405e+00  3.7437e-01   5.7175 1.081e-08 ***
## gift_count                   4.8235e-02  1.7918e-02   2.6919  0.007104 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Regression on donation amount

```
#Method1: exclude everyone who did not open original email
reg_amt<-lm(total_donation_amount~treat+opened_atleast_one_reminder+PGSS_age+last_gift_amount+treat_Ben
print(summary(reg_amt))
```

```
##
## Call:
## lm(formula = total_donation_amount ~ treat + opened_atleast_one_reminder +
##     PGSS_age + last_gift_amount + treat_Ben_inter + SYBUNT_Indicator +
##     LYBUNT_indicator + gift_count, data = compliant_orig)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.63 -10.86  -5.69   0.73 485.34
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  5.112637   4.431901   1.154   0.2490
## treat                        3.727689   4.007704   0.930   0.3526
## opened_atleast_one_reminder  0.761125   2.822980   0.270   0.7875
## PGSS_age                    -0.202273   0.179668  -1.126   0.2606
## last_gift_amount             0.009280   0.005257   1.765   0.0779 .
## treat_Ben_inter             -6.289508   4.895256  -1.285   0.1992
## SYBUNT_Indicator            -1.016602   3.880527  -0.262   0.7934
## LYBUNT_indicator             9.695540   3.320716   2.920   0.0036 **
## gift_count                   0.271003   0.211708   1.280   0.2009
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.65 on 844 degrees of freedom
##   (7 observations deleted due to missingness)
## Multiple R-squared:  0.03484,    Adjusted R-squared:  0.02569
## F-statistic: 3.808 on 8 and 844 DF,  p-value: 0.0002083
```

```
#Methods: all
#THis is intent to treat
reg_amt<-lm(total_donation_amount~treat+opened_atleast_one_reminder+PGSS_age+last_gift_amount+treat_Ben
print(summary(reg_amt))
```

```
##
## Call:
```

```
## lm(formula = total_donation_amount ~ treat + opened_atleast_one_reminder +
##     PGSS_age + last_gift_amount + treat_Ben_inter + SYBUNT_Indicator +
##     LYBUNT_indicator + gift_count, data = merged)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -634.0   -6.8   -0.5    3.6 3731.3
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -5.587919   5.260948  -1.062   0.2883
## treat                        10.250563   5.278876   1.942   0.0523 .
## opened_atleast_one_reminder   9.759698   4.719716   2.068   0.0388 *
## PGSS_age                      0.145309   0.255075   0.570   0.5690
## last_gift_amount              0.063043   0.005759  10.947   <2e-16 ***
## treat_Ben_inter             -10.658154   6.949666  -1.534   0.1253
## SYBUNT_Indicator             -7.205692   5.141771  -1.401   0.1612
## LYBUNT_indicator              1.395489   5.013627   0.278   0.7808
## gift_count                   -0.136168   0.391000  -0.348   0.7277
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 89.24 on 2089 degrees of freedom
##   (12 observations deleted due to missingness)
## Multiple R-squared:  0.06538,    Adjusted R-squared:  0.0618
## F-statistic: 18.27 on 8 and 2089 DF,  p-value: < 2.2e-16
```

Regression for effect of original resposne

```
#Add other regressions here

#Determining effect of two different splits on donation after original e-mail
just_msg_effect_immediate<-lm(donated_after_orig_email~Split.Name, data=merged)
print("Original treatment effect on immediate response:")
```

```
## [1] "Original treatment effect on immediate response:"
```

```
print(summary(just_msg_effect_immediate))
```

```
##
## Call:
## lm(formula = donated_after_orig_email ~ Split.Name, data = merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.01611 -0.01611 -0.00664 -0.00664  0.99336
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.016114   0.003263   4.938  8.5e-07 ***
## Split.NameSplit B -0.009479   0.004615  -2.054   0.0401 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.106 on 2108 degrees of freedom
## Multiple R-squared:  0.001997,   Adjusted R-squared:  0.001524
```

```
## F-statistic: 4.219 on 1 and 2108 DF,  p-value: 0.0401
```

Regression with HTE

```
#Regression for just the treatments and the last donation indicators
reg_last_donate_cat <- lm(donated_after_orig_email~Split.Name+merged$donated_TY_pre_treatment+LYBUNT_in
                          SYBUNT_Indicator+Never_donator, data=merged)
print("Regression of donations in response to first email against donation_TY_pre_treatment,LYBUNT, SYB
print(summary(reg_last_donate_cat))

#Regression for the treatments and the last donation indicators and treatments/indicators interactions
reg_last_donate_int <- lm(donated_after_orig_email~Split.Name+donated_TY_pre_treatment+LYBUNT_indicator
                          SYBUNT_Indicator+Never_donator+
                          Split.Name*donated_TY_pre_treatment+Split.Name*LYBUNT_indicator+Split.Name*S
                          Split.Name*Never_donator, data=merged)
print("Regression of treatments and last donation indicators as well as interactions")
print(summary(reg_last_donate_int))
```

Regression on donation delay

```
reg_delay<-lm(days_till_donation~treat+treat_reminder+opened_reminder1+opened_reminder2+PGSS.Year+Month
#print(summary(reg_delay))

sum(merged$total_donation_amount)
```

```
## [1] 11780.61
```

Calculate ITT