Student Dropout Prediction Challenge

Neha Bharambe

Introduction

Each year, almost 30% of the students drop out from their college education. This adversely affect not just the student but also universities in terms of resources, time and money.

Dropout prediction aids in uncovering potential risks which are often overlooked. Mitigating these risks will assist universities retain their students which in turn help students to complete their course work.

With this study, universities can beforehand predict students who are potential dropout candidates, so that they can proactively work with these students to resolve any challenges/issues they may have.

Objective

This project aims at predicting Student's success at completing his/her coursework. This implies whether a student will drop out from enrolled coursework or not.

Data

Data collected is for students pursuing Bachelor's degree during 2012 to 2017. This datasets is divided into three parts

1. Static Data

2. Progress Data

3. Financial Aid Data

Student Static Data

Static Data was collected for each student in the term they were enrolled. It contains demographic and educational background information about the students.

```
setwd("/Users/Nehu/StudentDropoutChallenge/Student Retention Challenge Data/S
tudent Static Data")
stFall2011 <- read.csv("Fall 2011_ST.csv",header = T)
stFall2012 <- read.csv("Fall 2012.csv",header = T)
stFall2013 <- read.csv("Fall 2013.csv",header = T)
stFall2014 <- read.csv("Fall 2014.csv",header = T)
stFall2015 <- read.csv("Fall 2015.csv",header = T)
stFall2016 <- read.csv("Fall 2016.csv",header = T)
stSpring2012 <- read.csv("Spring 2012_ST.csv",header = T)
```

```
stSpring2013 <- read.csv("Spring 2013.csv",header = T)
stSpring2014 <- read.csv("Spring 2014.csv",header = T)
stSpring2015 <- read.csv("Spring 2015.csv",header = T)
stSpring2016 <- read.csv("Spring 2016.csv",header = T)

studentStaticData <- rbind(stFall2011,stFall2012, stFall2013, stFall2014, stF
all2015, stFall2016, stSpring2012, stSpring2013, stSpring2014, stSpring2015,
stSpring2016)

head(studentStaticData)
##   StudentID  Cohort CohortTerm Campus           Address1 Address2
## 1    285848 2011-12          1     NA 328 Adams St Apt 1
## 2    302176 2011-12          1     NA      142 Cherry St
## 3    301803 2011-12          1     NA  12 Rainbow Street
## 4    302756 2011-12          1     NA   345 4th St Apt 2
## 5    300304 2011-12          1     NA      6600 Broadway   Apt 3D
## 6    301067 2011-12          1     NA         240 3rd St
##             City State  Zip RegistrationDate Gender BirthYear BirthMonth
## 1        Hoboken    NJ 7030         20110808      2      1978          9
## 2    Jersey City    NJ 7305         20110804      1      1970          4
## 3   Presque Isle    ME 4769         20110809      2      1984          4
## 4    Jersey City    NJ 7302         20110823      2      1986          1
## 5 West New York    NJ 7093         20110725      1      1992          2
## 6    Jersey City    NJ 7302         20110420      1      1969          4
##   Hispanic AmericanIndian Asian Black NativeHawaiian White TwoOrMoreRace
## 1        0              0     0     0              0     1             0
## 2        0              0     0     0              0     1             0
## 3        0              0     0     0              0     1             0
## 4        0              0     0     0              0     1             0
## 5        1              0     0     0              0     0             0
## 6        0              0     0     0              0     1             0
##   HSDip HSDipYr HSGPAUnwtd HSGPAWtd FirstGen DualHSSummerEnroll
## 1     1      -1      -1.00       -1       -1                  0
## 2     1      -1      -1.00       -1       -1                  0
## 3     1      -1      -1.00       -1       -1                  0
## 4    -1      -1      -1.00       -1       -1                  0
## 5     1    2010       3.13       -1       -1                  0
## 6     1      -1      -1.00       -1       -1                  0
##   EnrollmentStatus NumColCredAttemptTransfer NumColCredAcceptTransfer
## 1                2                         0                      0.0
## 2                2                        96                     45.0
## 3                2                         0                      0.0
## 4                2                        54                     87.5
## 5                1                        -2                     -2.0
## 6                2                        70                     66.0
##   CumLoanAtEntry HighDeg MathPlacement EngPlacement GatewayMathStatus
## 1             -1       0             0            0                 0
## 2             -1       0             0            0                 0
## 3             -1       0             0            0                 0
## 4             -1       0             0            0                 0
```

```
## 5             -2       0             1            0                  0
## 6             -1       2             0            0                  0
##   GatewayEnglishStatus
## 1                    0
## 2                    0
## 3                    0
## 4                    0
## 5                    0
## 6                    0
```

Student Progress data

Progress Data was collected for each student's activity for each term in each academic year. It contains Students' academic progression and outcomes over time. As it is collected for each academic year, I have Merged all the student progress data files to fetch progress data for latest Academic Year and corresponding latest term

```
library(RMySQL)
## Loading required package: DBI
mydb = dbConnect(MySQL(), user='root', password='*****', dbname='project')

rs <- dbSendQuery(mydb, "select b1.* from studentProgressData b1,(
select b.StudentID, b.academicYear, max(b.term) as maxterm from studentProgre
ssData b,
(select
StudentID
,max(AcademicYear) as y from
studentProgressData a
Group by a.StudentID)c
where b.studentid = c.studentid
and b.AcademicYear = c.y
group by b.StudentID, b.academicYear)x1
where x1.StudentID = b1.StudentID
and b1.AcademicYear = x1.AcademicYear
and b1.Term = x1.maxterm
;")
studentProgressData_max = dbFetch(rs, n = -1)

dbClearResult(rs)
## [1] TRUE
dbDisconnect(mydb)
## [1] TRUE
head(studentProgressData_max)
##   StudentID  Cohort CohortTerm Term AcademicYear CompleteDevMath
## 1    300412 2011-12          1    1      2011-12               0
## 2    303260 2011-12          1    1      2011-12              -2
## 3    304587 2011-12          1    1      2011-12              -2
## 4    305459 2011-12          1    1      2011-12              -1
## 5    303183 2011-12          1    1      2011-12              -2
## 6    305281 2011-12          1    1      2011-12              -2
```

```
##   CompleteDevEnglish  Major1 Major2 Complete1 Complete2 CompleteCIP1
## 1                 -2       0     -1         0         0           -2
## 2                 -2 13.1001     -1         0         0           -2
## 3                 -2 51.3801     -1         0         0           -2
## 4                 -1       0     -1         0         0           -2
## 5                  0 52.1401     -1         0         0           -2
## 6                 -2       0     -1         0         0           -2
##   CompleteCIP2 TransferIntent DegreeTypeSought TermGPA CumGPA
## 1           -2             -1                6    2.56   2.56
## 2           -2             -1                6    0.00   0.00
## 3           -2             -1                6    3.15   3.15
## 4           -2             -1                6    1.65   1.65
## 5           -2             -1                6    3.14   3.14
## 6           -2             -1                6    3.70   3.70
```

Financial Aid Data

Financial Aid Data was collected for each student for each academic year, and it is stored in different columns for different years. It contains Financial Aid and other related information such as scholarships, loans, gross income etc.

```
setwd("/Users/Nehu/StudentDropoutChallenge/Student Retention Challenge Data/S
tudent Financial Aid Data")
financialData <- read.csv("2011-2017_Cohorts_Financial_Aid_and_Fafsa_Data.csv
", header = TRUE)

head(financialData)
##   ID.with.leading  cohort cohort.term Marital.Status Adjusted.Gross.Income
## 1          297957 2011-12           1         Single                     0
## 2          302040 2011-12           1         Single                 18096
## 3          234532 2011-12           1         Single                 12383
## 4          303486 2011-12           1        Married                 59303
## 5          304316 2011-12           1         Single                 25133
## 6          302808 2011-12           1         Single                 15971
##   Parent.Adjusted.Gross.Income Father.s.Highest.Grade.Level
## 1                            0                      College
## 2                            0                  High School
## 3                            0                  High School
## 4                            0                  High School
## 5                            0                      Unknown
## 6                            0                Middle School
##   Mother.s.Highest.Grade.Level           Housing X2012.Loan
## 1                  High School On Campus Housing       3500
## 2                  High School        Off Campus      12500
## 3                  High School        Off Campus         NA
## 4                Middle School        Off Campus       4750
## 5                  High School                           NA
## 6                  High School        Off Campus       6500
##   X2012.Scholarship X2012.Work.Study X2012.Grant X2013.Loan
## 1                NA               NA       10714       5500
```

```
## 2                NA                NA        3500        6250
## 3                NA                NA        7432        5500
## 4                NA                NA         850        2750
## 5                NA                NA          NA          NA
## 6                NA                NA        5550        8000
##   X2013.Scholarship X2013.Work.Study X2013.Grant X2014.Loan
## 1                NA                NA       11095          NA
## 2                NA                NA          NA          NA
## 3                NA                NA          NA          NA
## 4                NA                NA        1650       10500
## 5                NA                NA          NA          NA
## 6                NA                NA        2888          NA
##   X2014.Scholarship X2014.Work.Study X2014.Grant X2015.Loan
## 1                NA                NA          NA          NA
## 2                NA                NA          NA          NA
## 3                NA                NA          NA          NA
## 4                NA                NA        3146        5206
## 5                NA                NA          NA          NA
## 6                NA                NA          NA          NA
##   X2015.Scholarship X2015.Work.Study X2015.Grant X2016.Loan
## 1                NA                NA          NA          NA
## 2                NA                NA          NA          NA
## 3                NA                NA          NA          NA
## 4                NA                NA        4580          NA
## 5                NA                NA          NA          NA
## 6                NA                NA          NA          NA
##   X2016.Scholarship X2016.Work.Study X2016.Grant X2017.Loan
## 1                NA                NA          NA          NA
## 2                NA                NA          NA          NA
## 3                NA                NA          NA          NA
## 4                NA                NA         691        8385
## 5                NA                NA          NA          NA
## 6                NA                NA          NA          NA
##   X2017.Scholarship X2017.Work.Study X2017.Grant
## 1                NA                NA          NA
## 2                NA                NA          NA
## 3                NA                NA          NA
## 4                NA                NA        2233
## 5                NA                NA          NA
## 6                NA                NA          NA
```

Training labels

List of Student IDs with dropout labels

```
setwd("/Users/Nehu/StudentDropoutChallenge")
TrainLabels <- read.csv("DropoutTrainLabels.csv", header = T)
```

Test IDs

List of student IDs for which prediction needs to be done.

```
setwd("/Users/Nehu/StudentDropoutChallenge/Student Retention Challenge Data/T
est Data")
testIds <- read.csv("TestIDs.csv", header = T)
```

Exploratory Data Analysis -

It reflects the descriptive statistics of variables in the financial aid dataset

Financial Aid Data

```
summary(financialData)
```

Variables	Min	1st Quartile	Median	Mean	3rd Quartile	Max	Missing values
Adjusted.Gross.Income	-24326	0	2637	13125	16323	2576425	2154
Parent.Adjusted.Gross.Income	-62979	0	12372	28102	38587	657631	2154
X2012.Loan	337	3500	5500	7169	9500	55626	12532
X2012.Scholarship	283	2000	4000	5225	6000	27632	13598
X2012.Work.Study	200	1700	2000	1873	2121	3000	13666
X2012.Grant	79.09	3368.25	5794	6660.93	10714	13263	12415
X2013.Loan	103	3500	5500	7156	9500	50555	11582
X2013.Scholarship	23	2000	3549	4793	6409	28737	13459
X2013.Work.Study	25	2000	2000	2084	2200	4000	13590
X2013.Grant	162	3683	6089	7094	11040	13790	11450
X2014.Loan	128	3783	6250	7280	10500	49845	11028
X2014.Scholarship	100	2000	4000	4999	6000	38851	13353
X2014.Work.Study	70	2000	2000	1933	2000	3300	13526
X2014.Grant	97.24	3528	6245	7208.11	11725.89	14001	10840
X2015.Loan	25	4162	6250	7241	10500	47824	10718
X2015.Scholarship	200	2000	4000	4755	5730	30478	13174
X2015.Work.Study	10	2000	2000	2127	2800	4600	13520
X2015.Grant	209	3880	6358	7370	11592	19038	10365
X2016.Loan	103	4500	6420	7625	10500	52880	10594
X2016.Scholarship	28.3	2000	4000	4897.3	6000	31265.5	13084
X2016.Work.Study	75	2000	2000	2036	2000	4000	13497
X2016.Grant	9.69	3963.25	6428	7458.96	11717.5	18505	10075
X2017.Loan	103	5354	6500	8256	11812	60118	10445
X2017.Scholarship	100	2000	4000	5024	6906	33848	12784

X2017.Work.Study	45	1500	2000	1929	2000	3000	13402
X2017.Grant	0.1	4261	7305	7794.2	12173	19823	9732

Cohort

2011-12	2012-13	2013-14	2014-15	2015-16	2016-17
2302	2267	2077	2244	2351	2528

Cohort Term

1	3
10667	3102

Housing

	Off Campus	On Campus Housing	With Parent
2164	5373	1624	4608

Marital Status

	Divorced	Married	Separated	Single
2154	236	1024	200	10155

Student Static Data

Summary reflects the descriptive statistics of the variables in the static dataset

```
summary(studentStaticData)
```

Variables	Min	1st Quartile	Median	Mean	3rd Quartile	Max	NA
BirthYear	1945	1986	1992	1989	1995	2000	1
Campus	0	0	0	0	0	0	13261
HSGPAUnwtd	-1	-1	-1	0.1624	2.4	4	
HSGPAWtd	-1	-1	-1	-1	-1	-1	
FirstGen	-1	-1	-1	-1	-1	-1	
DualHSSummerEnroll	0	0	0	0	0	0	
NumColCredAttemptTransfer	-2	-2	14	36.97	73	150	
NumColCredAcceptTransfer	-2	-2	22	31.77	66	96	
CumLoanAtEntry	-2	-2	-1	-1.41	-1	-1	

Variable	Missing	No	Yes
MathPlacement	571	8415	4275
EngPlacement	571	9640	3050
GatewayMathStatus	0	11673	1588
GatewayEnglishStatus	0	10739	2522
Hispanic	918	8020	4323
AmericanIndian	918	12319	24
Asian	918	11180	1163
Black	918	9506	2837
NativeHawaiian	918	12321	22
White	918	8998	3345
TwoOrMoreRace	918	12112	231

Enrollment Status

1	2
5452	7809

Gender

1- Male	2 - Female
5362	7899

Cohort

2011-12	2302
2012-13	2267
2013-14	2077
2014-15	2244
2015-16	2351
2016-17	2020

Cohort Term

1	3
10667	2594

Highest Degree

0	2	3	4
9463	3639	157	2

Student Progress Data

Summary reflects the descriptive statistics of the variables in the Progress dataset

```
summary(studentProgressData_max)
```

Variables	Min	1st Quartile	Median	Mean	3rd Quartile	Max
TermGPA	0	1.725	3.08	2.592	3.7	4
CumGPA	0	2.3	3.07	2.778	3.58	4

CompleteCIP1	-2	-2	-2	10.52	23.01	54.01
CompleteCIP2	-2	-2	-2	-2	-2	-2

Transfer Intent

-1	13767

DegreeTypeSought

6	13767

Complete1

0	7	8
10035	1209	2523

Term

1	3270
3	8266
6	2231

Data Cleaning for Financial Aid data

Majority of students are single and leaving Off campus, therefore imputing the empty values of Marital Status and Housing with the majority

```
financialData$Marital.Status <- sub("^$", "Single", financialData$Marital.Sta
tus)

financialData$Housing <- sub("^$", "Off Campus", financialData$Housing)
```

Imputing the empty values of parent's Highest Grade level with 'Unknown'.

```
financialData$Father.s.Highest.Grade.Level <- sub("^$", "Unknown", financialD
ata$Father.s.Highest.Grade.Level)
financialData$Mother.s.Highest.Grade.Level <- sub("^$", "Unknown", financialD
ata$Mother.s.Highest.Grade.Level)
library(imputeTS)
financialData <- na.replace(financialData, 0)
```

Data Cleaning for Student Static Data

```
#All the values for Campus variable are missing for all students, not signifi
cant in analysis
studentStaticData$Campus <- NULL

#Imputing the missing value with mean for birth year
studentStaticData$BirthYear <- na.replace(studentStaticData$BirthYear, 1989)
```

```
#Converting the different columns of ethnicity to one row for simplicity of a
nalysis
for (i in (1:nrow(studentStaticData))){
  if(studentStaticData$Hispanic[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'Hispanic'
  } else if (studentStaticData$AmericanIndian[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'AmericanIndian'
  } else if (studentStaticData$Asian[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'Asian'
  } else if ( studentStaticData$Black[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'Black'
  } else if ( studentStaticData$NativeHawaiian[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'NativeHawaiian'
  } else if ( studentStaticData$White[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'White'
  } else if ( studentStaticData$TwoOrMoreRace[i] == 1) {
    studentStaticData$Ethnicity[i] <- 'TwoOrMoreRace'
  } else {
    studentStaticData$Ethnicity[i] <- 'Unknown'
  }
}

studentStaticData$Ethnicity <- as.factor(studentStaticData$Ethnicity)


#Missing values for HSDip is imputed as 1,and the reason is to get a admissio
n in college, student requires high school completion certificate

studentStaticData$HSDip <- ifelse(studentStaticData$HSDip == -1, NA, studentS
taticData$HSDip)
studentStaticData$HSDip <- na.replace(studentStaticData$HSDip, 1)


#Imputing themissing value of HSGPAUnwtd as zero
studentStaticData$HSGPAUnwtd <- ifelse(studentStaticData$HSGPAUnwtd == -1, NA
, studentStaticData$HSGPAUnwtd)
studentStaticData$HSGPAUnwtd <- na.replace(studentStaticData$HSGPAUnwtd, 0)

#All values of HSGPAWtd, FirstGen are missing, removing the column for analys
is
#All values of DualHSSummerEnroll are 0, which means Not past dual enrollment
nor summer enrollee, removing column for analysis
studentStaticData$HSGPAWtd <- NULL
studentStaticData$FirstGen <- NULL
studentStaticData$DualHSSummerEnroll <- NULL


#Imputed the missing values to zero of credit attempt transfer
studentStaticData$NumColCredAttemptTransfer <- ifelse((studentStaticData$NumC
```

```
olCredAttemptTransfer == -1), NA, studentStaticData$NumColCredAttemptTransfer
)
studentStaticData$NumColCredAttemptTransfer <- na.replace(studentStaticData$N
umColCredAttemptTransfer, 0)


#Imputed the missing values to zero of credit attempt transfer
studentStaticData$NumColCredAcceptTransfer <- ifelse((studentStaticData$NumCo
lCredAcceptTransfer == -1), NA, studentStaticData$NumColCredAcceptTransfer)
studentStaticData$NumColCredAcceptTransfer <- na.replace(studentStaticData$Nu
mColCredAcceptTransfer, 0)

#CumLoanAtEntry
#All the values are missing or unknown, removing column for analysis
studentStaticData$CumLoanAtEntry <- NULL
```

Data Cleaning for Student Progress Data

```
#Imputing the missing values Major1 as zero
studentProgressData_max$Major1 <- as.numeric(studentProgressData_max$Major1)
studentProgressData_max$Major1 <- ifelse(studentProgressData_max$Major1 == -1
, NA, studentProgressData_max$Major1)
studentProgressData_max$Major1 <- na.replace(studentProgressData_max$Major1,
0)

#All the values for Complete2 are zero, removing the column for analysis
studentProgressData_max$Complete2 <- NULL

#All the values for CIP2 are unknown, so removing the column for analysis
studentProgressData_max$CompleteCIP2 <- NULL

#All the values for TransferIntent are missing, so removing the column for an
alysis
studentProgressData_max$TransferIntent <- NULL

#All students are pursuing bachelor's degree, so removing the column for anal
ysis
studentProgressData_max$DegreeTypeSought <- NULL
```

Merge financial data, static data and progress data

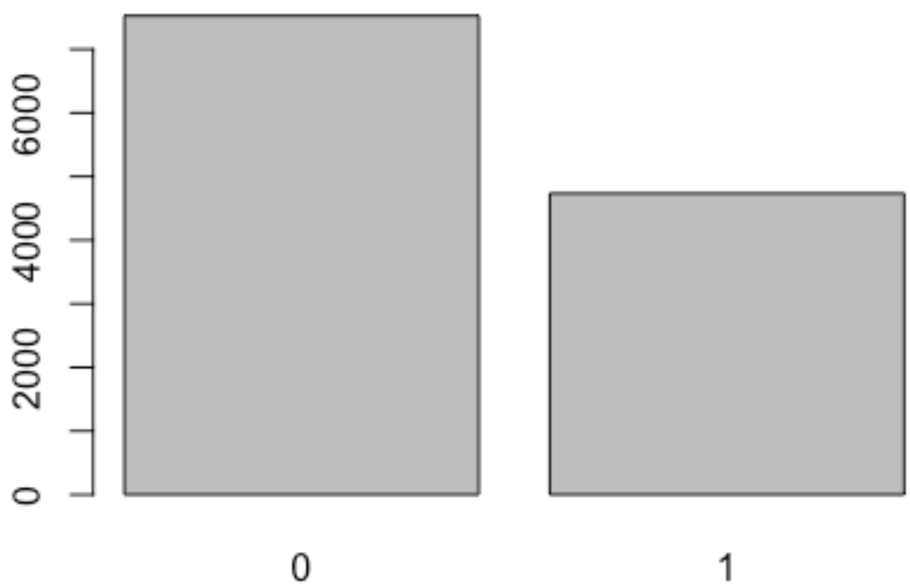
```
combinedStudentData <- merge(x = studentProgressData_max, y = studentStaticDa
ta,by = c("StudentID","Cohort","CohortTerm"))
financialData.static.progress <- merge(x = combinedStudentData, y = financial
Data,
                                       by.y = "ID.with.leading", by.x = "Stud
entID")
```

Merge the train labels with combined dataset

```
CombinedData.trainLabels <- merge(x = TrainLabels, y = financialData.static.p
rogress,
                                  by.y = "StudentID", by.x = "StudentID")

CombinedData.trainLabels$Dropout <- as.factor(CombinedData.trainLabels$Dropou
t)

barplot(table(CombinedData.trainLabels$Dropout))
```



Dropout

Did not Dropout	Drop out
7527	4734
61%	39%

From above table it is clear that it is balanced dataset.

Approach for Prediction Model:

Split dataset into training and testing

To avoid overfitting and to check the robustness of model, we will divide the complete dataset into 2 parts with proportion of 75% - training and remaining 25% - testing and will monitor the model performance by 5-fold cross validation.

```
library(caret)
## Loading required package: lattice
## Loading required package: ggplot2
set.seed(31)
intrain <- createDataPartition(CombinedData.trainLabels$Dropout,p=0.75,list =
FALSE)
train1 <- CombinedData.trainLabels[intrain,]
test1 <- CombinedData.trainLabels[-intrain,]
trctrl <- trainControl(method = "cv", number = 5)
```

Methodology:

All the variables are used for model building except the address information and the information which was common for all the students.

Model Type: Classification Tree

```
model1 <- train(Dropout ~ Cohort + CohortTerm + Gender + BirthYear + BirthMon
th + HSDipYr
                + HSGPAUnwtd + EnrollmentStatus + Ethnicity + HSDip + HSGPAUn
wtd + NumColCredAttemptTransfer + NumColCredAcceptTransfer
                + HighDeg + MathPlacement + EngPlacement
                + GatewayMathStatus + GatewayEnglishStatus
                + Marital.Status + Adjusted.Gross.Income + Parent.Adjusted.Gr
oss.Income
                + Father.s.Highest.Grade.Level + Mother.s.Highest.Grade.Level
                + Housing + X2012.Loan + X2012.Scholarship + X2012.Work.Study
+ X2012.Grant
                + X2013.Loan + X2013.Scholarship + X2013.Work.Study + X2013.G
rant
                + X2014.Loan + X2014.Scholarship + X2014.Work.Study + X2014.G
rant
                + X2015.Loan + X2015.Scholarship + X2015.Work.Study + X2015.G
rant
                + X2016.Loan + X2016.Scholarship + X2016.Work.Study + X2016.G
rant
                + X2017.Loan + X2017.Scholarship + X2017.Work.Study + X2017.G
rant
                + Term + AcademicYear + CompleteDevEnglish
                + CompleteDevMath + Major2 + CompleteCIP1
                + Major1 + Complete1
```

```
                + TermGPA + CumGPA
                , data = train1, method = "rpart", trControl=trctrl)

predictions1 <- predict(model1, newdata = test1)

confusionMatrix(predictions1, test1$Dropout)$overall[1]
##  Accuracy
## 0.9484334
bagImp1 <- varImp(model1, scale=TRUE)
```

Accuracy for Classification Tree – 94.84%

Important variables:

```
bagImp1
## rpart variable importance
##
##   only 20 most important variables shown (out of 248)
##
##                                Overall
## CompleteCIP1                  100.0000
## Complete18                     58.8271
## AcademicYear2016-17            37.3503
## CumGPA                         35.5380
## X2017.Grant                    20.6703
## TermGPA                        20.3983
## Complete17                     15.0213
## Cohort2015-16                   3.4173
## X2012.Grant                     2.8008
## X2016.Loan                      2.3115
## X2013.Grant                     1.9388
## Parent.Adjusted.Gross.Income    1.8429
## Cohort2016-17                   1.6642
## X2012.Loan                      1.6562
## X2016.Grant                     1.5215
## X2013.Loan                      0.9316
## X2014.Grant                     0.7588
## X2016.Scholarship               0.6158
## X2015.Scholarship               0.5915
## X2017.Loan                      0.4937
```

Model Type: Kth Nearest Neighbor

```
model2 <- train(Dropout ~ Cohort + CohortTerm + Gender + BirthYear + BirthMon
th + HSDipYr
                + HSGPAUnwtd + EnrollmentStatus + Ethnicity + HSDip + HSGPAUn
wtd + NumColCredAttemptTransfer + NumColCredAcceptTransfer
                + HighDeg + MathPlacement + EngPlacement
                + GatewayMathStatus + GatewayEnglishStatus
```

```
                + Marital.Status + Adjusted.Gross.Income + Parent.Adjusted.Gr
oss.Income
                + Father.s.Highest.Grade.Level + Mother.s.Highest.Grade.Level
                + Housing + X2012.Loan + X2012.Scholarship + X2012.Work.Study
+ X2012.Grant
                + X2013.Loan + X2013.Scholarship + X2013.Work.Study + X2013.G
rant
                + X2014.Loan + X2014.Scholarship + X2014.Work.Study + X2014.G
rant
                + X2015.Loan + X2015.Scholarship + X2015.Work.Study + X2015.G
rant
                + X2016.Loan + X2016.Scholarship + X2016.Work.Study + X2016.G
rant
                + X2017.Loan + X2017.Scholarship + X2017.Work.Study + X2017.G
rant
                + Term + AcademicYear + CompleteDevEnglish
                + CompleteDevMath + Major2 + CompleteCIP1
                + Major1 + Complete1
                + TermGPA + CumGPA
                , data = train1, method = "knn", trControl=trctrl)


predictions2 <- predict(model2, newdata = test1)
confusionMatrix(predictions2, test1$Dropout)$overall[1]
##  Accuracy
## 0.7859008
bagImp2 <- varImp(model2, scale=TRUE)
```

Accuracy for KNN – 78.59%

Important variables:

```
bagImp2
## ROC curve variable importance
##
##   only 20 most important variables shown (out of 57)
##
##                                Importance
## AcademicYear                       100.00
## CompleteCIP1                        71.84
## Complete1                           71.71
## Cohort                              69.38
## TermGPA                             66.91
## X2017.Grant                         62.40
## CumGPA                              61.38
## X2017.Loan                          46.02
## Term                                43.24
## Major1                              37.45
## HSDipYr                             32.01
## X2016.Grant                         30.62
```

```
## BirthYear                        29.47
## X2016.Loan                       23.91
## Father.s.Highest.Grade.Level     18.63
## X2017.Scholarship                18.63
## EnrollmentStatus                 15.49
## CohortTerm                       15.18
## NumColCredAcceptTransfer         13.75
## X2012.Grant                      13.55
```

Model Type: Bagging

```
model4 <- train(Dropout ~ Cohort + CohortTerm + Gender + BirthYear + BirthMon
th + HSDipYr
                + HSGPAUnwtd + EnrollmentStatus + Ethnicity + HSDip + HSGPAUn
wtd + NumColCredAttemptTransfer + NumColCredAcceptTransfer
                + HighDeg + MathPlacement + EngPlacement
                + GatewayMathStatus + GatewayEnglishStatus
                + Marital.Status + Adjusted.Gross.Income + Parent.Adjusted.Gr
oss.Income
                + Father.s.Highest.Grade.Level + Mother.s.Highest.Grade.Level
                + Housing + X2012.Loan + X2012.Scholarship + X2012.Work.Study
+ X2012.Grant
                + X2013.Loan + X2013.Scholarship + X2013.Work.Study + X2013.G
rant
                + X2014.Loan + X2014.Scholarship + X2014.Work.Study + X2014.G
rant
                + X2015.Loan + X2015.Scholarship + X2015.Work.Study + X2015.G
rant
                + X2016.Loan + X2016.Scholarship + X2016.Work.Study + X2016.G
rant
                + X2017.Loan + X2017.Scholarship + X2017.Work.Study + X2017.G
rant
                + Term + AcademicYear + CompleteDevEnglish
                + CompleteDevMath + Major2 + CompleteCIP1
                + Major1 + Complete1
                + TermGPA + CumGPA
                , data = train1, method = "treebag", trControl=trctrl)

predictions4 <- predict(model4, newdata = test1)
confusionMatrix(predictions4, test1$Dropout)$overall[1]
##  Accuracy
## 0.9562663
bagImp4 <- varImp(model4, scale=TRUE)
```

Accuracy for Bagging – 95.62%

Important variables:

```
bagImp4
## treebag variable importance
##
##   only 20 most important variables shown (out of 261)
##
##                                Overall
## CompleteCIP1                   100.000
## Complete18                      55.931
## AcademicYear2016-17             37.381
## CumGPA                          34.040
## TermGPA                         28.517
## X2017.Grant                     23.919
## Complete17                      15.341
## Cohort2016-17                    4.900
## Parent.Adjusted.Gross.Income     4.599
## X2016.Loan                       3.930
## X2016.Grant                      3.797
## X2012.Grant                      3.449
## BirthMonth                       3.385
## NumColCredAttemptTransfer        3.279
## Cohort2015-16                    3.124
## NumColCredAcceptTransfer         2.838
## X2013.Grant                      2.732
## X2017.Loan                       2.328
## Adjusted.Gross.Income            2.296
## X2015.Loan                       2.235
```

Model Type: Logistic Regression

```
model5 <- train(Dropout ~ Cohort + CohortTerm + Gender + BirthYear + BirthMon
th + HSDipYr
                + HSGPAUnwtd + EnrollmentStatus + Ethnicity + HSDip + HSGPAUn
wtd + NumColCredAttemptTransfer + NumColCredAcceptTransfer
                + HighDeg + MathPlacement + EngPlacement
                + GatewayMathStatus + GatewayEnglishStatus
                + Marital.Status + Adjusted.Gross.Income + Parent.Adjusted.Gr
oss.Income
                + Father.s.Highest.Grade.Level + Mother.s.Highest.Grade.Level
                + Housing + X2012.Loan + X2012.Scholarship + X2012.Work.Study
+ X2012.Grant
                + X2013.Loan + X2013.Scholarship + X2013.Work.Study + X2013.G
rant
                + X2014.Loan + X2014.Scholarship + X2014.Work.Study + X2014.G
rant
                + X2015.Loan + X2015.Scholarship + X2015.Work.Study + X2015.G
rant
                + X2016.Loan + X2016.Scholarship + X2016.Work.Study + X2016.G
rant
                + X2017.Loan + X2017.Scholarship + X2017.Work.Study + X2017.G
```

```
rant
                + Term + AcademicYear + CompleteDevEnglish
                + CompleteDevMath + Major2 + CompleteCIP1
                + Major1 + Complete1
                + TermGPA + CumGPA
                , data = train1, method = "glm", family="binomial", trControl
=trctrl)
predictions5 <- predict(model5, newdata = test1)
confusionMatrix(predictions5, test1$Dropout)$overall[1]
##  Accuracy
## 0.9500653
bagImp5 <- varImp(model5, scale=TRUE)
```

Accuracy for Logistic Regression – 95%

Important variables:

```
bagImp5
## glm variable importance
##
##   only 20 most important variables shown (out of 234)
##
##                  Overall
## `Cohort2016-17` 1.000e+02
## BirthYear1949   1.429e-04
## BirthYear2000   1.429e-04
## HSDipYr1988     4.719e-05
## HSDipYr1993     4.719e-05
## HSDipYr1996     4.713e-05
## HSDipYr1980     4.713e-05
## HSDipYr1970     4.713e-05
## HSDipYr2000     4.713e-05
## HSDipYr1991     4.713e-05
## HSDipYr1979     4.713e-05
## HSDipYr2003     4.713e-05
## HSDipYr1998     4.713e-05
## HSDipYr1984     4.713e-05
## Complete17      5.119e-07
## Complete18      4.630e-07
## `Cohort2015-16` 4.371e-07
## X2016.Grant     3.453e-07
## X2016.Loan      2.294e-07
## `Cohort2014-15` 2.188e-07
```

Model Type: Model Stacking with random forest

```
#Construct data frame with predictions
predDF <- data.frame(predictions1,predictions2, predictions4, class = test1$D
ropout)
predDF$class <- as.factor(predDF$class)
#Combine models using random forest
```

```
combModFit.rf <- train(class ~ .
                       , method = "rf", data = predDF, distribution = 'binomi
al')
## note: only 2 unique complexity parameters in default grid. Truncating the
grid to 2 .
combPred.rf <- predict(combModFit.rf, predDF)
confusionMatrix(combPred.rf, predDF$class)$overall[1]
##  Accuracy
## 0.9562663
```

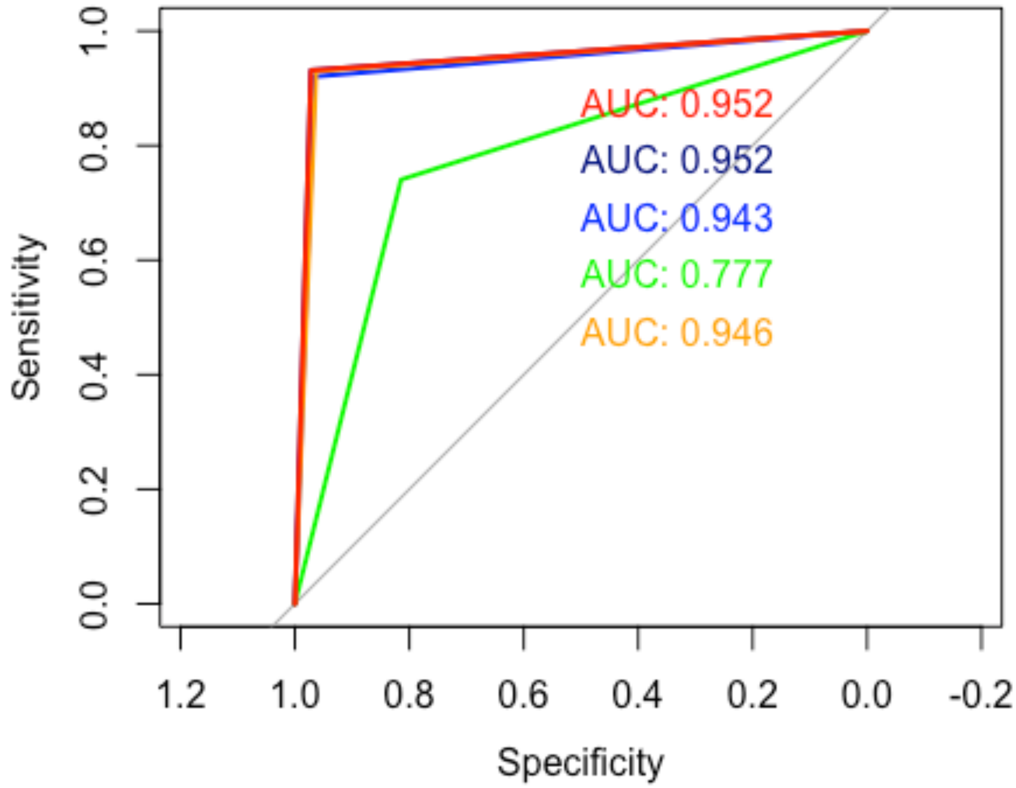
Accuracy for Model Stacking – 95.62%

ROC curve

ROC is a probability curve and It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s.

```
library(pROC)
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
roccurve1 <- roc(test1$Dropout ~ as.numeric(predictions1))
roccurve2 <- roc(test1$Dropout ~ as.numeric(predictions2))
roccurve4 <- roc(test1$Dropout ~ as.numeric(predictions4))
roccurve5 <- roc(test1$Dropout ~ as.numeric(predictions5))


#ROC Curve for model stacking
roccurve <- roc(predDF$class ~ as.numeric(combPred.rf))
roccurve$auc
## Area under the curve: 0.9517
roccurve$sensitivities
## [1] 1.00000 0.93153 0.00000
roccurve$specificities
## [1] 0.0000000 0.9718235 1.0000000
plot(roccurve1, print.auc = TRUE, col = "blue",print.auc.y = .7)
plot(roccurve2, print.auc = TRUE,
     col = "green", print.auc.y = .6, add = TRUE)
plot(roccurve4, print.auc = TRUE,
     col = "navy blue", print.auc.y = .8, add = TRUE)
plot(roccurve5, print.auc = TRUE,
     col = "orange", print.auc.y = .5, add = TRUE)
plot(roccurve, print.auc = TRUE,
     col = "red", print.auc.y = .9, add = TRUE)
```



From the graph, it is seen that the AUC of model stacking and bagging is better than other models.

Results

```
financialData.static.testIDs <- merge(x = testIds, y = financialData.static.p
rogress,
                                       by.y = "StudentID", by.x = "StudentID")
predictions1 <- predict(model1, newdata = financialData.static.testIDs)
predictions2 <- predict(model2, newdata = financialData.static.testIDs)
predictions4 <- predict(model4, newdata = financialData.static.testIDs)


test_predDF <- data.frame( predictions1, predictions2, predictions4)

test_combPred.rf <- predict(combModFit.rf,newdata = test_predDF)
```

When we run the prediction model on TestIDs, Accuracy is 95.91% (Kaggle)

Conclusion

Prediction of student's study success is possible through information collected by universities. Therefore, they can predict accurately whether a student will drop out or not.

Regarding the methodology, data wrangling and feature analysis plays a crucial role in model selection. The prediction accuracy of both Bagging and Model stacking is similar.

Downsides of Model stacking is it is complex and difficult to interpret whereas Bagging is less complicated and easy to interpret.