Student Loan Defaults

Bo Suzow

August 28, 2018

Introduction

Student loan debt (SLD) total in the US reached a staggering number in 1.5 trillion dollars, borrowed by 44 million people. It is 2.4 times larger than the total of credit card debt. The SLD total in 2008 was 640 billion dollars which ballooned to 1.2 trillion by 2015 (https://www.politifact.com/truth-o-meter/statements/2015/aug/14/jeb-bush/jeb-bush-student-loan-debt-has-doubled-under-obama/).

While [the return on investment of higher education (HED) is well known and documented] (https://college-education.procon.org/ (https://college-education.procon.org/)), student loans accumulated for financing higher education are reported as societal issues such as reasons for divorce (https://www.yahoo.com/amphtml/finance/news/millennial-marriages-crumbling-student-loan-debt-134145853.html), financial dependence on parents, and lack of home ownership observed in Gen-Y.

In order for a student to be able to pay off HED loans, the loan total should not exceed his/her annual income from gainful employment the HED would provide. For unfortunate some, this rule of thumb was violated and loan defaults resulted.

The US Department of Education publishes the College Scorecard data to help the public to make informed decisions about investments in higher education. The data features large amount of metrics including default rates and is organized by academic year. In this report, we will focus on the 2014-15 scorecard data. The goals of this report are:

- Explore the data to ascertain some trends the data is telling us.
- Find out strong predictors affecting default rates.
- Build a predictive model for default rates.

Data Load

This data set is a part of a data bundle (https://ed-public-download.app.cloud.gov/downloads/CollegeScorecard_Raw_Data.zip) published by the US Department of Education. This analysis utilizes the data pertinent to the 2014-15 academic year cohort (`MERGED2014_15_PP.csv`). The full data documentation is found here (https://collegescorecard.ed.gov/assets/FullDataDocumentation.pdf).

```
# Load the college scorecard data set for academic year 14-15

scorecard1415 <- read.csv("MERGED2014_15_PP.csv",na.strings=c("NULL","PrivacySuppressed"))
scorecard1415 <- tbl_df(scorecard1415)
```

Data Wrangling

Strategy 1

- Clean up the data set.
    - Select variables relevant to this study.
    - Standardize column names if necessary.
    - Inspect presence of missing value.
    - Determin missing value treatment strategy if applicable.

- Transform data types appropriately.

Select relevant variables

Two issues with the data set are noticed immediately.

- First, the number of variables/features is overwhelmingly large at 1700+.
- Secondly, many rows have missing values.

To address the first issue, the data element list (https://collegescorecard.ed.gov/assets/CollegeScorecardDataDictionary.xlsx) has been reviewed. Using the data documentation (https://collegescorecard.ed.gov/assets/FullDataDocumentation.pdf) as guidelines, the variables are selected in the following code block:

```
sc1415.all <- scorecard1415 %>% select(OPEID6,INSTNM,STABBR,NUMBRANCH,CONTROL,PREDDEG,
                                REGION,
                                LOCALE,LATITUDE,LONGITUDE,CCBASIC,
                                CCUGPROF,CCSIZSET,RELAFFIL,
                                ADM_RATE_ALL,
                                DISTANCEONLY,UGDS,UG,
                                UGDS_WHITE,UGDS_BLACK,UGDS_HISP,UGDS_ASIAN,
                                CURROPER,
                                NPT4_PUB,NPT4_PRIV,
                                NUM4_PUB,NUM4_PRIV,
                                TUITFTE,INEXPFTE,AVGFACSAL,
                                PFTFAC,
                                PCTPELL,
                                C150_4,
                                RET_FT4,
                                PCTFLOAN,
                                UG25ABV,
                                CDR2,
                                CDR3,
                                PAR_ED_PCT_1STGEN,
                                DEP_INC_AVG,
                                IND_INC_AVG,
                                DEBT_MDN,
                                GRAD_DEBT_MDN,
                                WDRAW_DEBT_MDN,
                                FAMINC,
                                MD_FAMINC,
                                POVERTY_RATE,
                                MN_EARN_WNE_P10,
                                MD_EARN_WNE_P10,
                                CDR3_DENOM
)
```

The `CDR3` column reports the default rate. Hence, the observations with missing values in the column cannot be used for this analysis. They are removed.

```
sc1415 <- sc1415.all %>% filter(!is.na(CDR3))
```

As for missing values, tt's observed that they are rather concentrated on the schools classified as stand alone graduate institutions. Let's remove them from the analysis.

```
sc1415 <- sc1415 %>% filter(PREDDEG!=4)
```

Missing Values

Even after removing the grad school rows, are there any columns with missiong values?

```
names(sc1415)[colSums(is.na(sc1415))>0]
```

```
##  [1] "LOCALE"            "LATITUDE"          "LONGITUDE"
##  [4] "CCBASIC"           "CCUGPROF"          "CCSIZSET"
##  [7] "RELAFFIL"          "ADM_RATE_ALL"      "DISTANCEONLY"
## [10] "UGDS"              "UG"                "UGDS_WHITE"
## [13] "UGDS_BLACK"        "UGDS_HISP"         "UGDS_ASIAN"
## [16] "CURROPER"          "NPT4_PUB"          "NPT4_PRIV"
## [19] "NUM4_PUB"          "NUM4_PRIV"         "TUITFTE"
## [22] "INEXPFTE"          "AVGFACSAL"         "PFTFAC"
## [25] "PCTPELL"           "C150_4"            "RET_FT4"
## [28] "PCTFLOAN"          "UG25ABV"           "CDR2"
## [31] "PAR_ED_PCT_1STGEN" "DEP_INC_AVG"       "IND_INC_AVG"
## [34] "DEBT_MDN"          "GRAD_DEBT_MDN"     "WDRAW_DEBT_MDN"
## [37] "FAMINC"            "MD_FAMINC"         "POVERTY_RATE"
## [40] "MN_EARN_WNE_P10"   "MD_EARN_WNE_P10"
```

44 of 50 columns still have missing values. Let's compute their proportions and select those under 10%.

```
naProportion <- apply(sc1415,2,function(x){sum(is.na(x))/nrow(sc1415)})
naProportion[naProportion<.1]
```

```
##            OPEID6             INSTNM            STABBR         NUMBRANCH
##       0.000000000       0.000000000       0.000000000       0.000000000
##           CONTROL           PREDDEG            REGION      DISTANCEONLY
##       0.000000000       0.000000000       0.000000000       0.063054647
##              UGDS        UGDS_WHITE        UGDS_BLACK         UGDS_HISP
##       0.063988790       0.063988790       0.063988790       0.063988790
##         UGDS_ASIAN           TUITFTE          INEXPFTE           PCTPELL
##       0.063988790       0.064611552       0.064611552       0.065390005
##          PCTFLOAN              CDR3 PAR_ED_PCT_1STGEN       DEP_INC_AVG
##       0.065390005       0.000000000       0.076911101       0.035964503
##       IND_INC_AVG          DEBT_MDN     GRAD_DEBT_MDN    WDRAW_DEBT_MDN
##       0.035964503       0.015569049       0.062120504       0.084072863
##            FAMINC         MD_FAMINC        CDR3_DENOM
##       0.003580881       0.003580881       0.000000000
```

Let's confirm that all `CDR3` and `CDR3_DENOM` values are numeric.

```
!is.numeric(sc1415$CDR3)
```

```
## [1] FALSE
```

```
!is.numeric(sc1415$CDR3_DENOM)
```

```
## [1] FALSE
```

Variable selection

Now select the variables with less than 10% missing value. Further clean up by eliminating the rows with missing values (Caveat: This is a simpliest approach of treating missing values. This may have to be revisited in pursuit of a more efficient strategy.)

(Qu: a better way to select the variables of interest without hard coding, but utilizing the output from the `naProportion` code chunk?)

```
sc1415 <- sc1415 %>% select(OPEID6, INSTNM, STABBR, NUMBRANCH, CONTROL, PREDDEG, REGION,
                            DISTANCEONLY, TUITFTE, INEXPFTE,
                            PCTPELL, PCTFLOAN, CDR3, PAR_ED_PCT_1STGEN,
                            DEP_INC_AVG, IND_INC_AVG,
                            DEBT_MDN, GRAD_DEBT_MDN, WDRAW_DEBT_MDN,
                            FAMINC, MD_FAMINC, CDR3_DENOM)

sc1415.net <- sc1415[complete.cases(sc1415),]


names(sc1415)[colSums(is.na(sc1415.net))>0]
```

```
## character(0)
```

The total number of rows has reduced from 6423 to 5210. The summary statistics of default rates (`CDR3`) are comparable amongst the three data sets – with graduate schools present, without grad schools, and without missing values. We will move ahead with the the `sc1415.net` data frame for the analysis. Please note that the unclassified institution (`PREDDEG = 0`) have been removed at the missing-value row elimination.

(Qu: Should box plots be added to confirm this point?)

```
summary(sc1415.all$CDR3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0000  0.0600  0.1160  0.1235  0.1770  0.7890    1016
```

```
summary(sc1415$CDR3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0650  0.1200  0.1272  0.1800  0.7890
```

```
summary(sc1415.net$CDR3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.069   0.125   0.129   0.182   0.789
```

Strategy 2

Stragegy 1 left us with a few number of numeric variables. A quick exploratory plots show no so strong associations between them and default rates.

Armed with some knowledge Stragety 1 revealed, let's take a different approach:

- Remove all rows with missing `CDR3` values. It reduces the # of variables from 7100+ to 1700+.
- Remove stand-alone gradudate institutions and those unclassified (`PREDDEG == 0 or 4`). _ Remove all columns with missing values.

```
sc1415.cdr3 = scorecard1415 %>% filter(!is.na(CDR3))
sc1415.cdr3 = sc1415.cdr3 %>% filter(!(PREDDEG == 0 | PREDDEG == 4))
sc1415.cdr3 <- sc1415.cdr3[,colSums(is.na(sc1415.cdr3))==0]
```

This results in 6018 rows and 247 columns. This data frame is bigger than one resulted from Strategy 1, its features are mostly reporting fields of study (Classification of Instructional Programs). We will move ahead with the Strategy 1's data frame `sc1415.net`, but keep `sc1415.cdr3` for complimentary data if needed.

Converting to factor variables

Convert `CONTROL`, `PREDDEG`, `DISTANCEONLY`, and `REGION` to factor variables.

```
control_list <- c(1:3)
control_descs <- c("Public",
                   "Private nonprofit",
                   "Private for-profit")

sc1415.net <- sc1415.net %>% mutate(CONTROL = factor(CONTROL,levels=control_list,
                                            labels=control_descs))
# 1 Public
# 2 Private nonprofit
# 3 Private for-profit


preddeg_list <- c(1:3)
preddeg_descs <- c(
                   "Certificate",
                   "Associate's",
                   "Bachelor's"
                   )

sc1415.net <- sc1415.net %>% mutate(PREDDEG = factor(PREDDEG,levels=preddeg_list,
                                            labels=preddeg_descs))

# 0 Not classified  IPEDS  -- not included in the analysis
# 1 Predominantly certificate-degree granting
# 2 Predominantly associate's-degree granting
# 3 Predominantly bachelor's-degree granting
# 4 Entirely graduate-degree granting -- not included in the analysis


distanceonly_list = c(0:1)
distanceonly_descs = c("Not Online-Ed Only",
                   "Online-Ed Only")

sc1415.net <- sc1415.net %>% mutate(DISTANCEONLY = factor(DISTANCEONLY,levels=distanceonly_list,
                                            labels=distanceonly_descs))

#0  Not distance-education only
#1  Distance-education only


region_list <- c(0:9)
region_descs <- c("U.S. Service Schools",
                  "New England",
                  "Mid East",
                  "Great Lakes",
                  "Plains",
                  "Southeast",
                  "Southwest",
                  "Rocky Mtn",
                  "Far West",
                  "Outlying Areas")

sc1415.net <- sc1415.net %>% mutate(REGION = factor(REGION,levels=region_list,
                                            labels=region_descs))


# 0 U.S. Service Schools
# 1 New England (CT, ME, MA, NH, RI, VT)
# 2 Mid East (DE, DC, MD, NJ, NY, PA)
```

```
# 3 Great Lakes (IL, IN, MI, OH, WI)
# 4 Plains (IA, KS, MN, MO, NE, ND, SD)
# 5 Southeast (AL, AR, FL, GA, KY, LA, MS, NC, SC, TN, VA, WV)
# 6 Southwest (AZ, NM, OK, TX)
# 7 Rocky Mountains (CO, ID, MT, UT, WY)
# 8 Far West (AK, CA, HI, NV, OR, WA)
# 9 Outlying Areas (AS, FM, GU, MH, MP, PR, PW, VI)
```

Exploratory Data Analysis (WIP)

Quick Descriptive Statistics

The total number of students who are in repayment as of FYR 2014-15 is 34.6 million. Of these, 31 million are the students from certificate, associate's or bachelor's degree programs. Their average default rate is 0.1290004 with the standard deviation of 0.0748101.

Total count, mean and standard deviation of default rates (`CDR3`) by `CONTROL` - ownership type.

```
table(sc1415.net$CONTROL)
```

```
##
##             Public  Private nonprofit Private for-profit
##               1509               1191               2510
```

```
sc1415.net %>% group_by(CONTROL) %>% summarize(mean(CDR3),sd(CDR3))
```

```
## # A tibble: 3 x 3
##   CONTROL            `mean(CDR3)` `sd(CDR3)`
##   <fct>                     <dbl>      <dbl>
## 1 Public                   0.135      0.0775
## 2 Private nonprofit        0.0773     0.0632
## 3 Private for-profit       0.150      0.0663
```

By `PREDDEG` - predominant degree awarded.

```
table(sc1415.net$PREDDEG)
```

```
##
## Certificate Associate's  Bachelor's
##        2109        1298        1803
```

```
sc1415.net %>% group_by(PREDDEG) %>% summarize(mean(CDR3),sd(CDR3))
```

```
## # A tibble: 3 x 3
##   PREDDEG     `mean(CDR3)` `sd(CDR3)`
##   <fct>              <dbl>      <dbl>
## 1 Certificate       0.151      0.0725
## 2 Associate's       0.164      0.0657
## 3 Bachelor's        0.0783     0.0538
```

By `DISTANCEONLY` - whether distance education only or not.

```
table(sc1415.net$DISTANCEONLY)
```

```
##
## Not Online-Ed Only     Online-Ed Only
##               5181                 29
```

```
sc1415.net %>% group_by(DISTANCEONLY) %>% summarize(mean(CDR3),sd(CDR3))
```

```
## # A tibble: 2 x 3
##   DISTANCEONLY       `mean(CDR3)` `sd(CDR3)`
##   <fct>                     <dbl>      <dbl>
## 1 Not Online-Ed Only        0.129     0.0748
## 2 Online-Ed Only            0.127     0.0830
```

By `REGION` - geographical location.

```
table(sc1415.net$REGION)
```

```
##
## U.S. Service Schools          New England             Mid East
##                    0                  295                  806
##          Great Lakes               Plains            Southeast
##                  839                  483                 1282
##            Southwest            Rocky Mtn             Far West
##                  546                  201                  696
##       Outlying Areas
##                   62
```

```
sc1415.net %>% group_by(REGION) %>% summarize(mean(CDR3),sd(CDR3))
```

```
## # A tibble: 9 x 3
##   REGION         `mean(CDR3)` `sd(CDR3)`
##   <fct>                 <dbl>      <dbl>
## 1 New England          0.0950     0.0643
## 2 Mid East             0.103      0.0633
## 3 Great Lakes          0.126      0.0709
## 4 Plains               0.123      0.0704
## 5 Southeast            0.144      0.0782
## 6 Southwest            0.155      0.0713
## 7 Rocky Mtn            0.142      0.0727
## 8 Far West             0.129      0.0809
## 9 Outlying Areas       0.133      0.0705
```

Summaries of numerical variables.

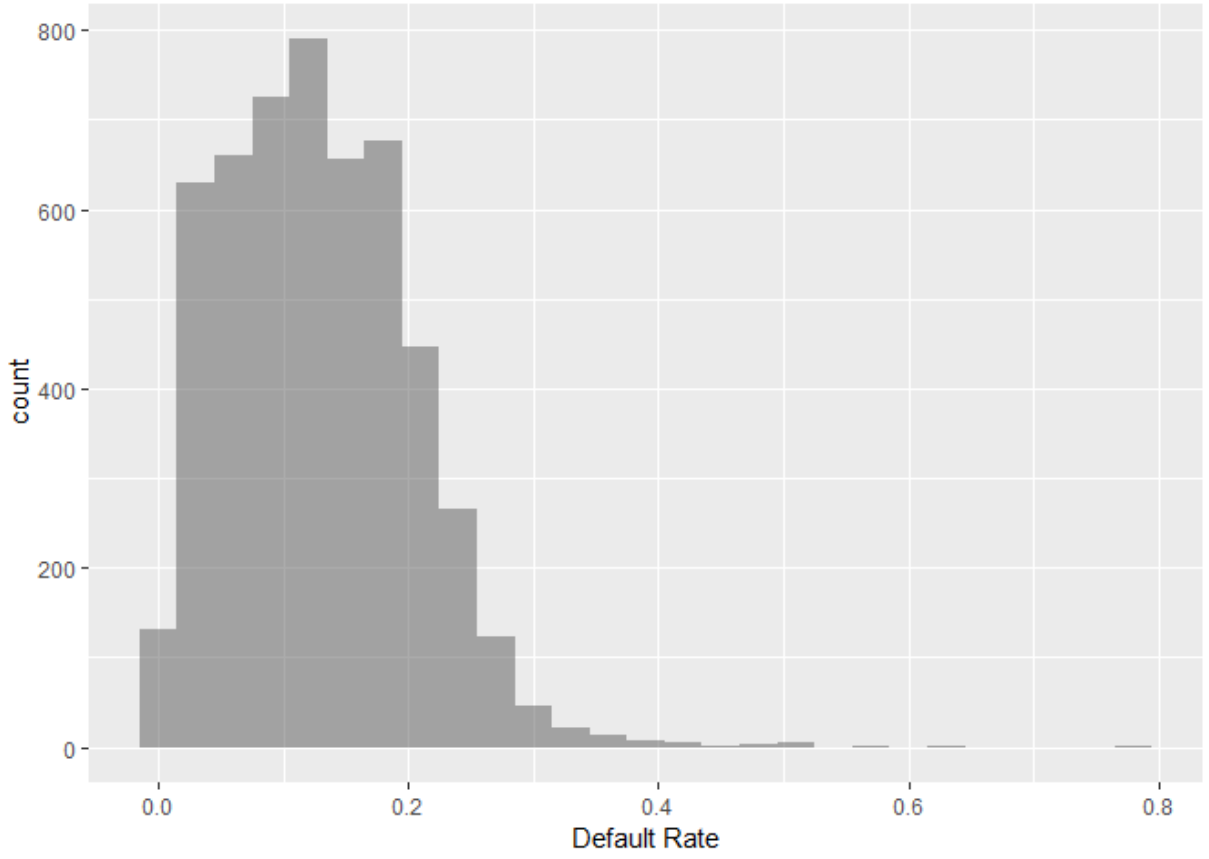
```
sc1415.net %>% group_by(CONTROL) %>% summarize(mean(CDR3),sd(CDR3))
```

```
## # A tibble: 3 x 3
##   CONTROL            `mean(CDR3)` `sd(CDR3)`
##   <fct>                     <dbl>      <dbl>
## 1 Public                   0.135      0.0775
## 2 Private nonprofit        0.0773     0.0632
## 3 Private for-profit       0.150      0.0663
```

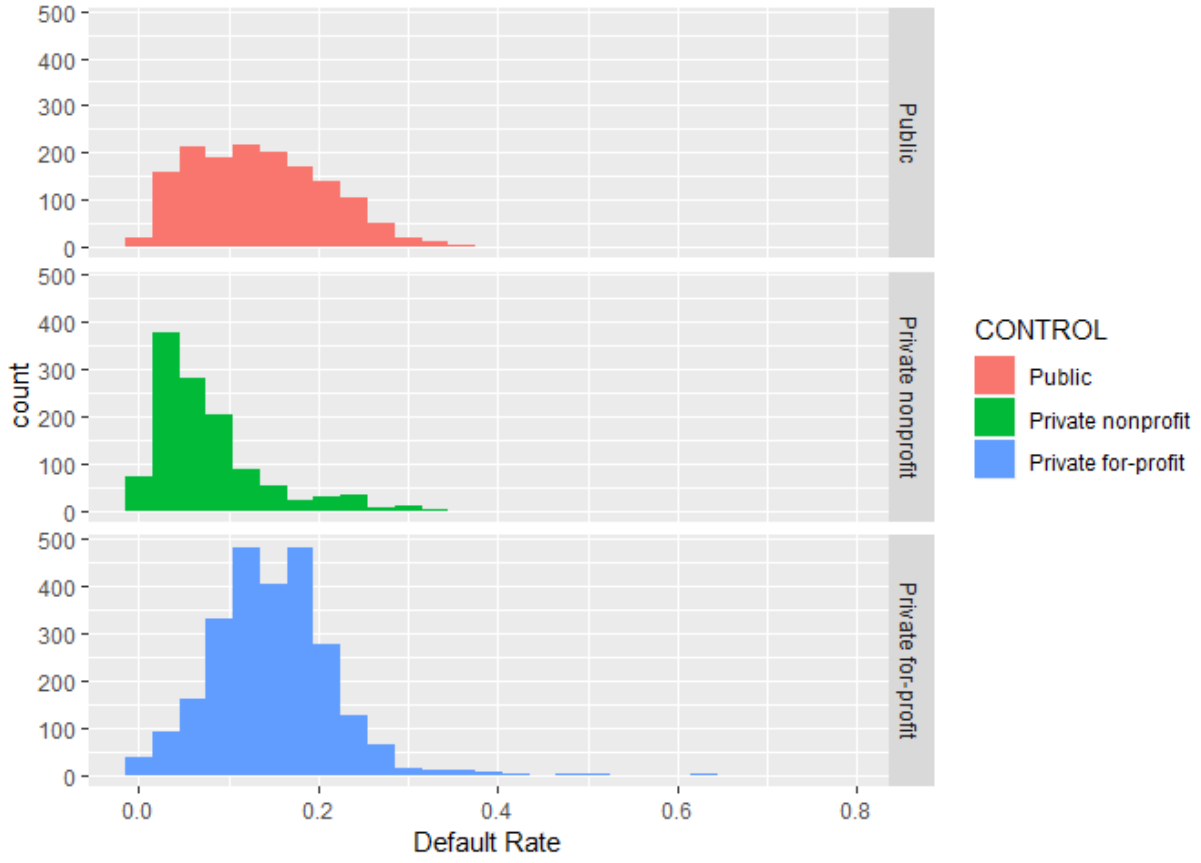
Plotting

Let's draw some histograms.

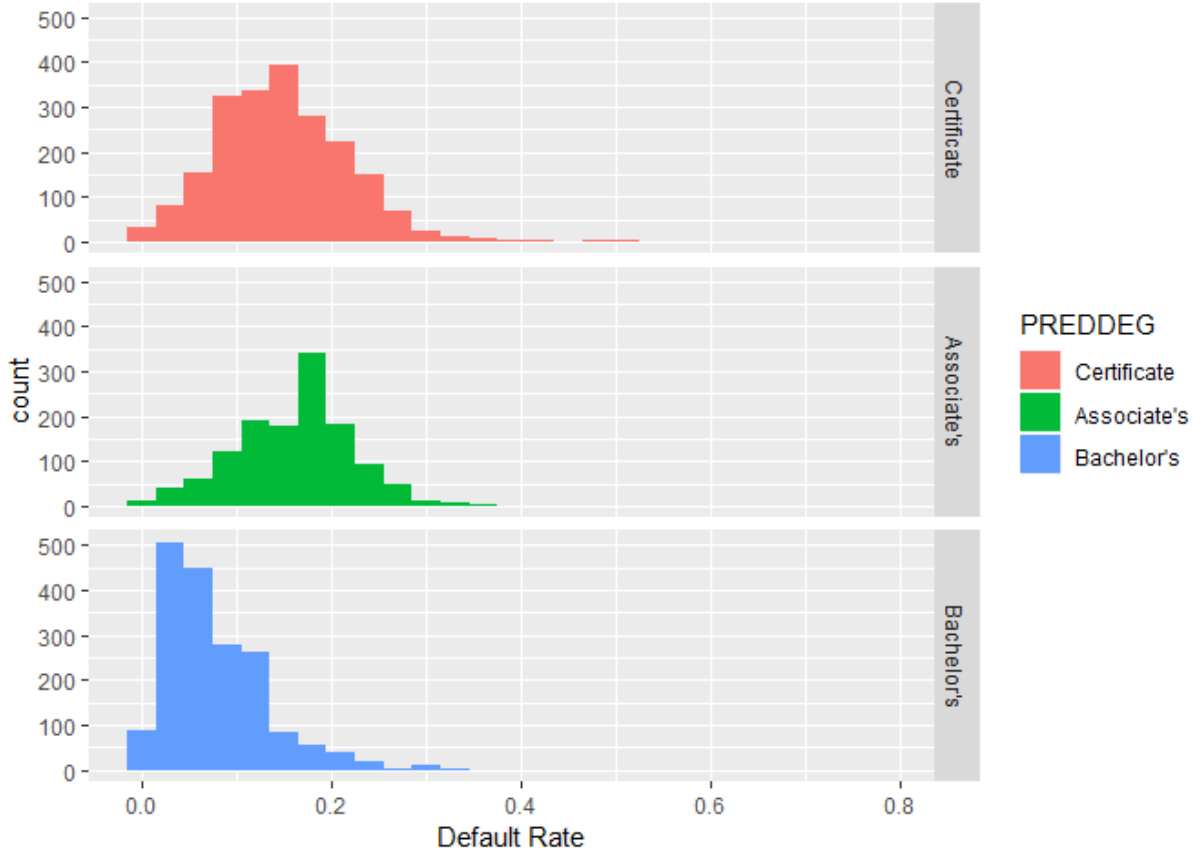
```
ggplot(sc1415.net,aes(x=CDR3)) +
   geom_histogram(binwidth=.03,alpha=.5) +
   xlab("Default Rate")
```



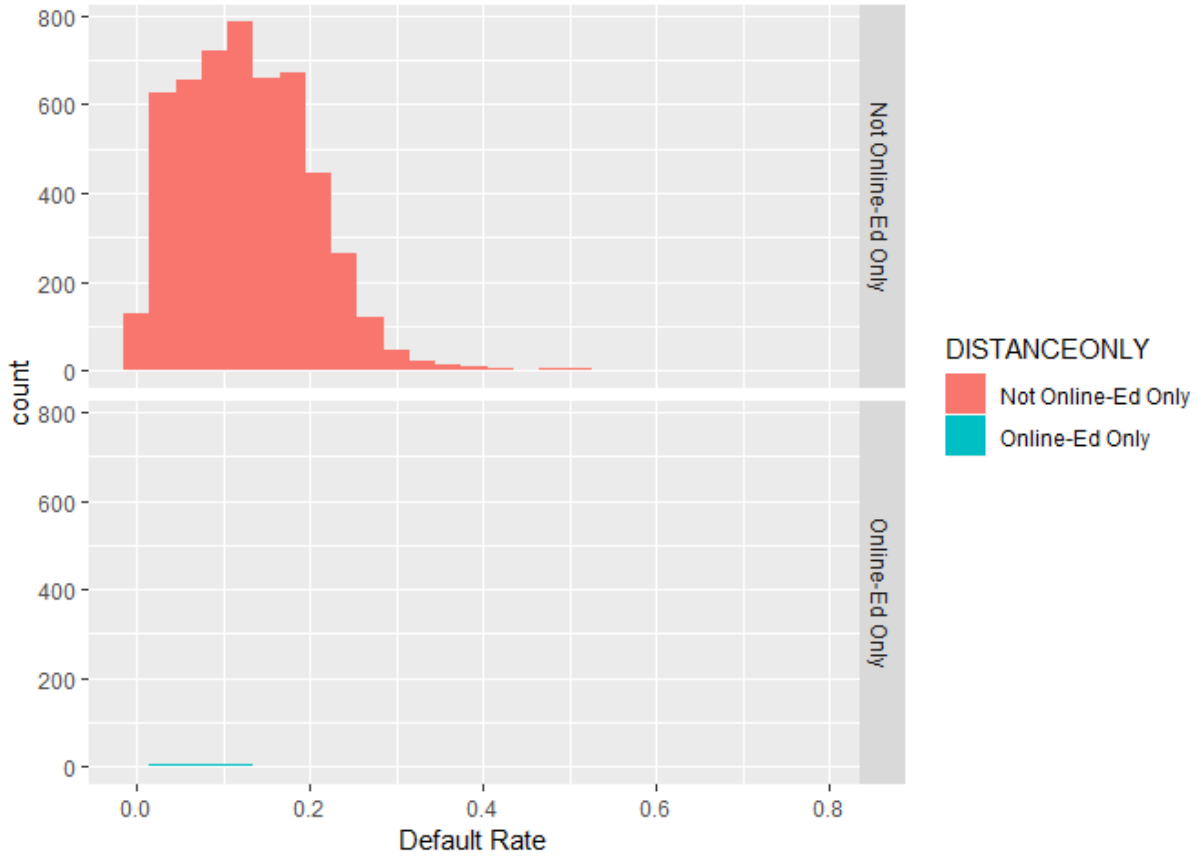

```
ggplot(sc1415.net,aes(x=CDR3, fill=CONTROL)) +
   geom_histogram(binwidth=.03) +
   facet_grid(CONTROL~.) +
   xlab("Default Rate")
```



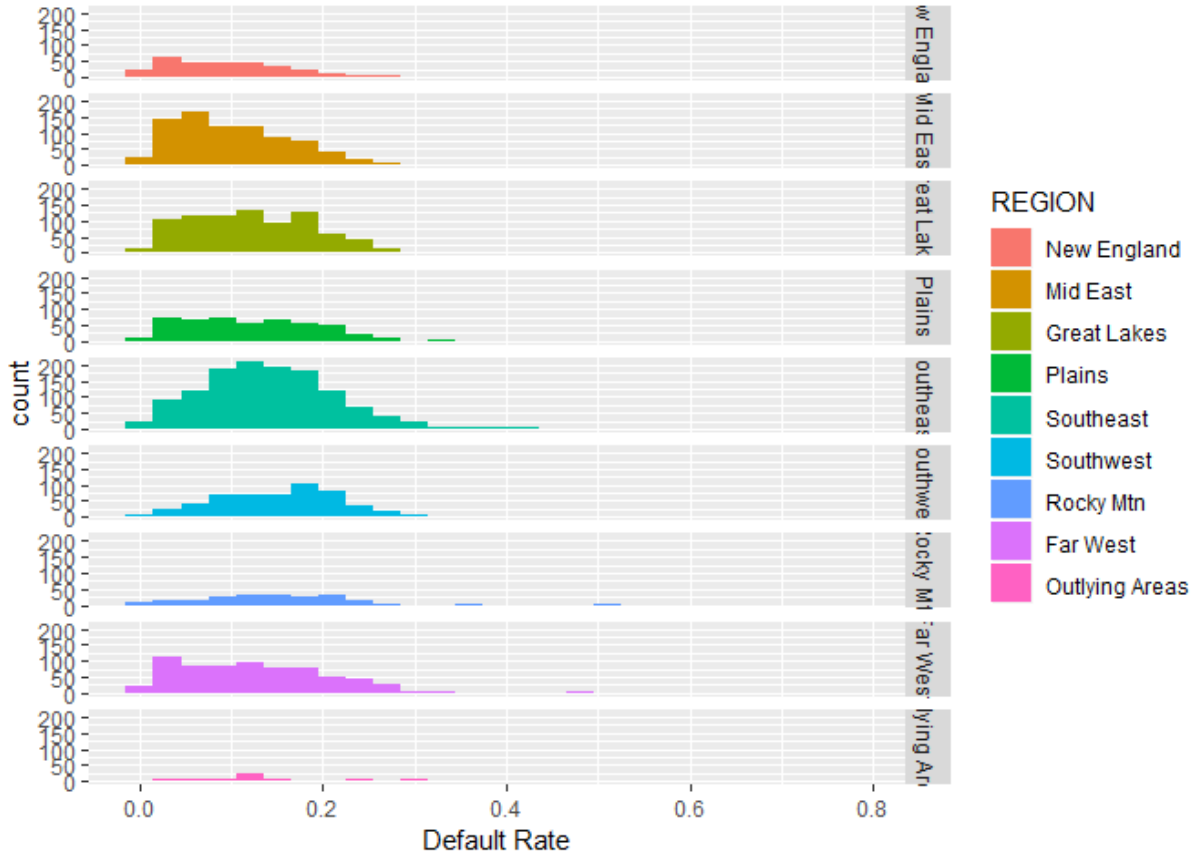

```
ggplot(sc1415.net,aes(x=CDR3, fill=PREDDEG)) +
   geom_histogram(binwidth=.03) +
   facet_grid(PREDDEG~.) +
   xlab("Default Rate")
```



```
ggplot(sc1415.net,aes(x=CDR3, fill=DISTANCEONLY)) +
   geom_histogram(binwidth=.03) +
   facet_grid(DISTANCEONLY~.) +
   xlab("Default Rate")
```



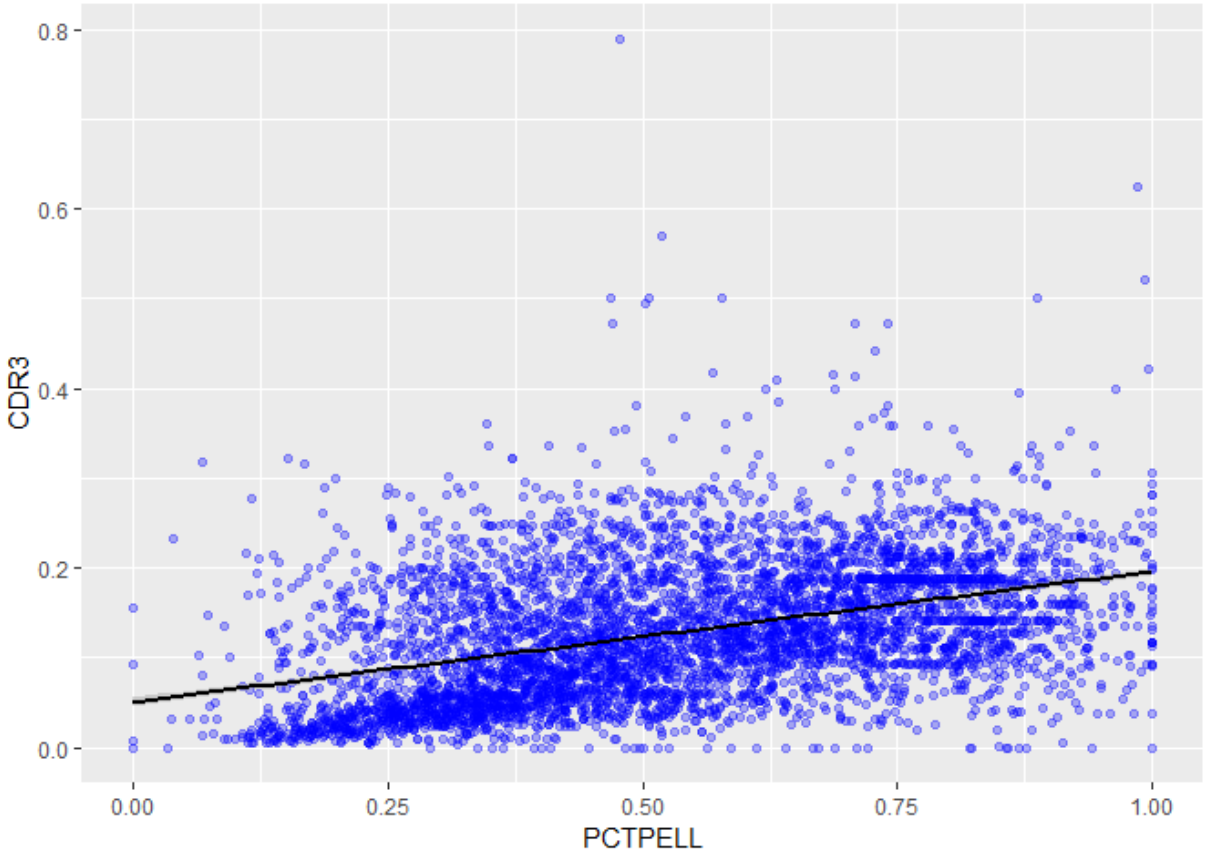

```
ggplot(sc1415.net,aes(x=CDR3, fill=REGION)) +
   geom_histogram(binwidth=.03) +
   facet_grid(REGION~.) +
   xlab("Default Rate")
```



Scatter Plots

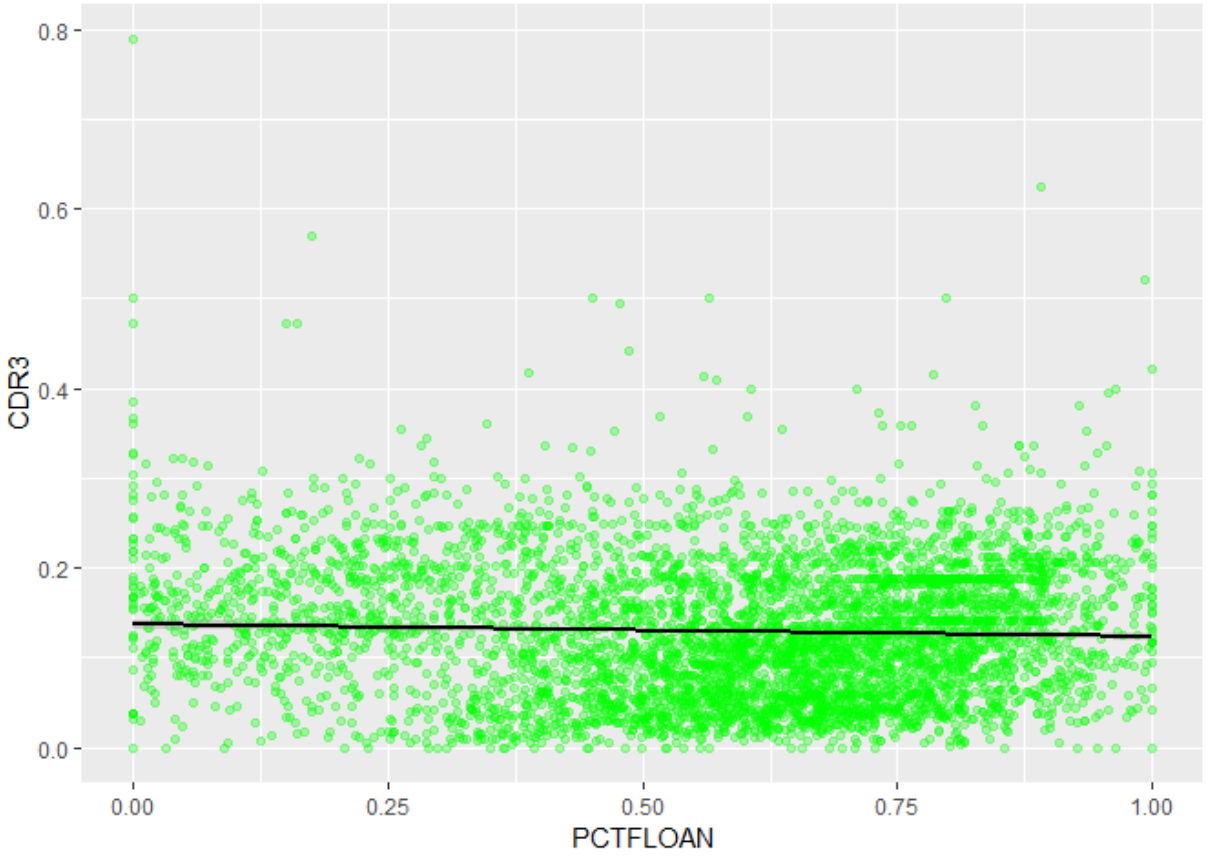
Percent of Pell Grant vs default rate

```
ggplot(sc1415.net,aes(x=PCTPELL,y=CDR3)) +
   geom_point(alpha=.3, col='Blue') +
   geom_smooth(method="lm", col="black")
```



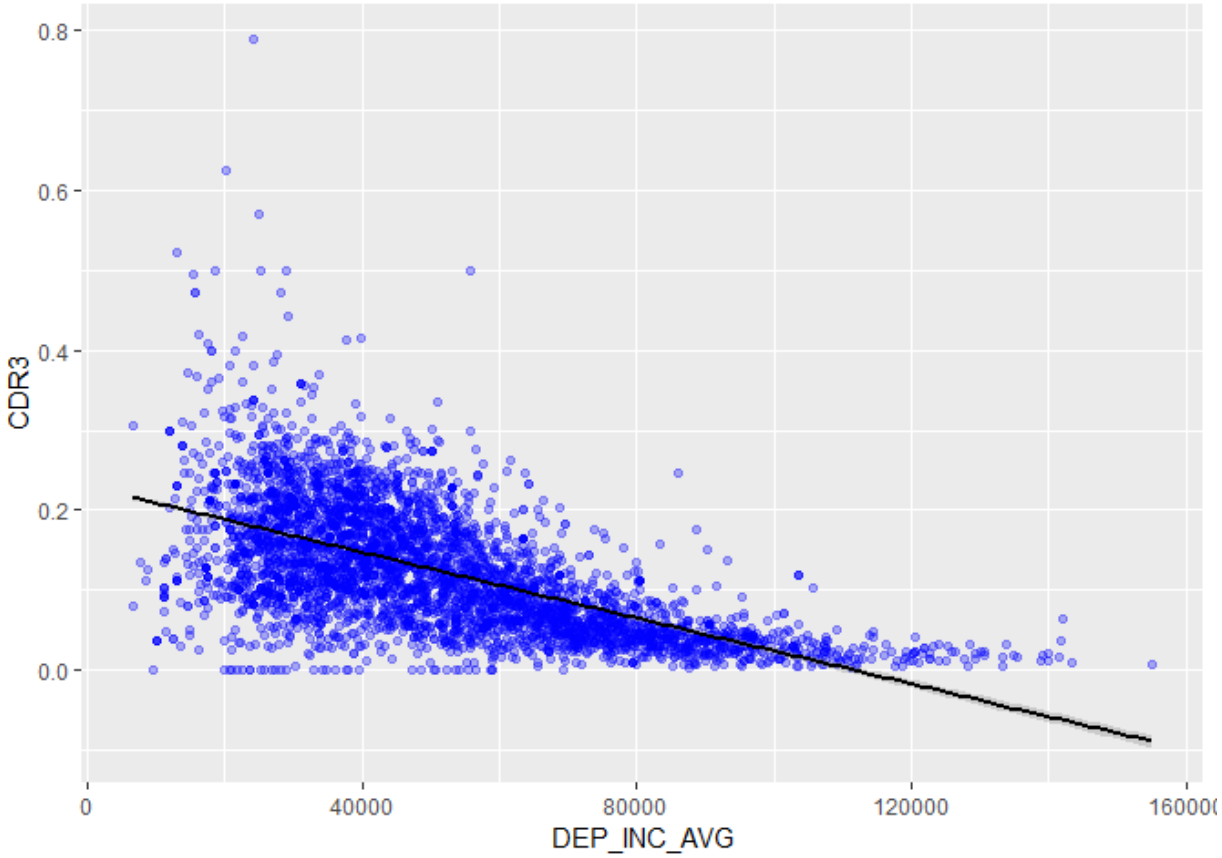
Percent of Federal loan vs default rate

```
ggplot(sc1415.net,aes(x=PCTFLOAN,y=CDR3)) +
   geom_point(alpha=.3, col='green') +
   geom_smooth(method="lm", col="black")
```



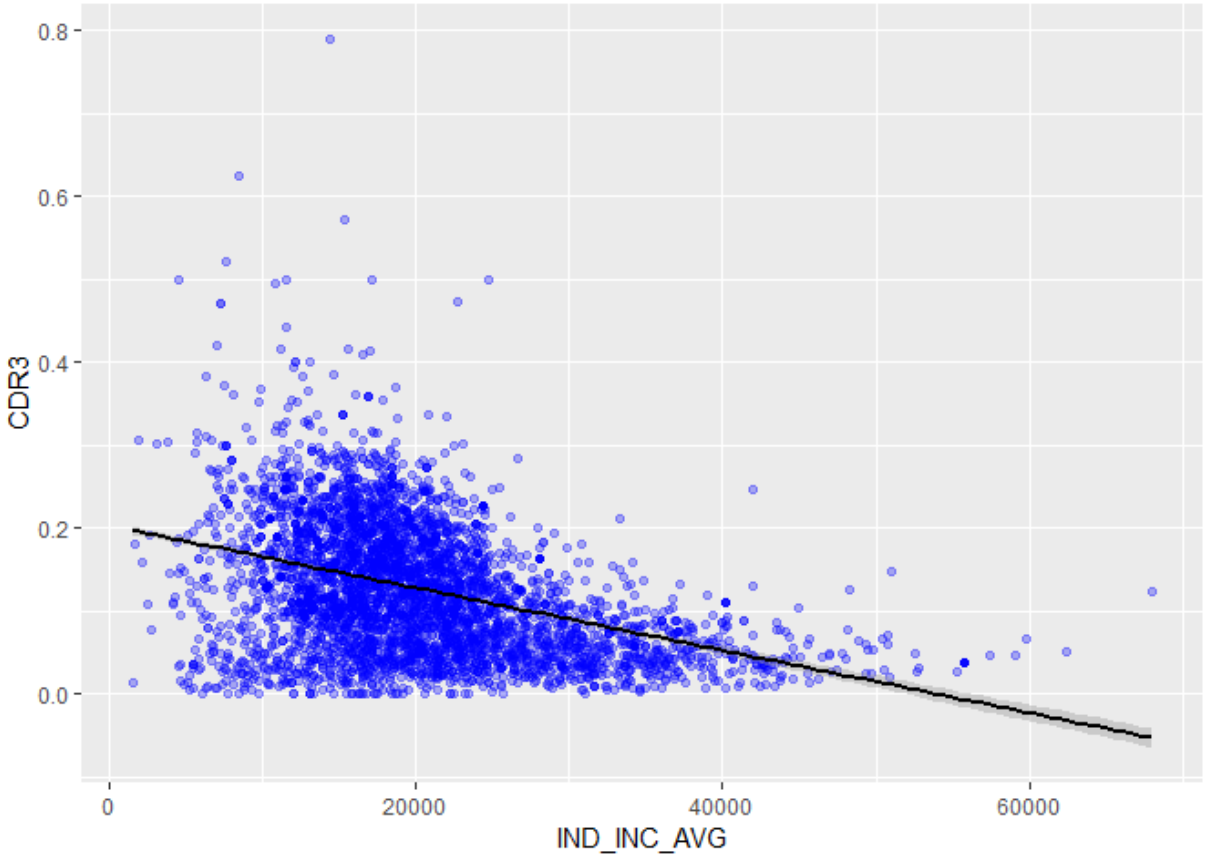
Average family income of dependent students vs default rate

```
ggplot(sc1415.net,aes(x=DEP_INC_AVG,y=CDR3)) +
   geom_point(alpha=.3, col='blue') +
   geom_smooth(method="lm", col="black")
```



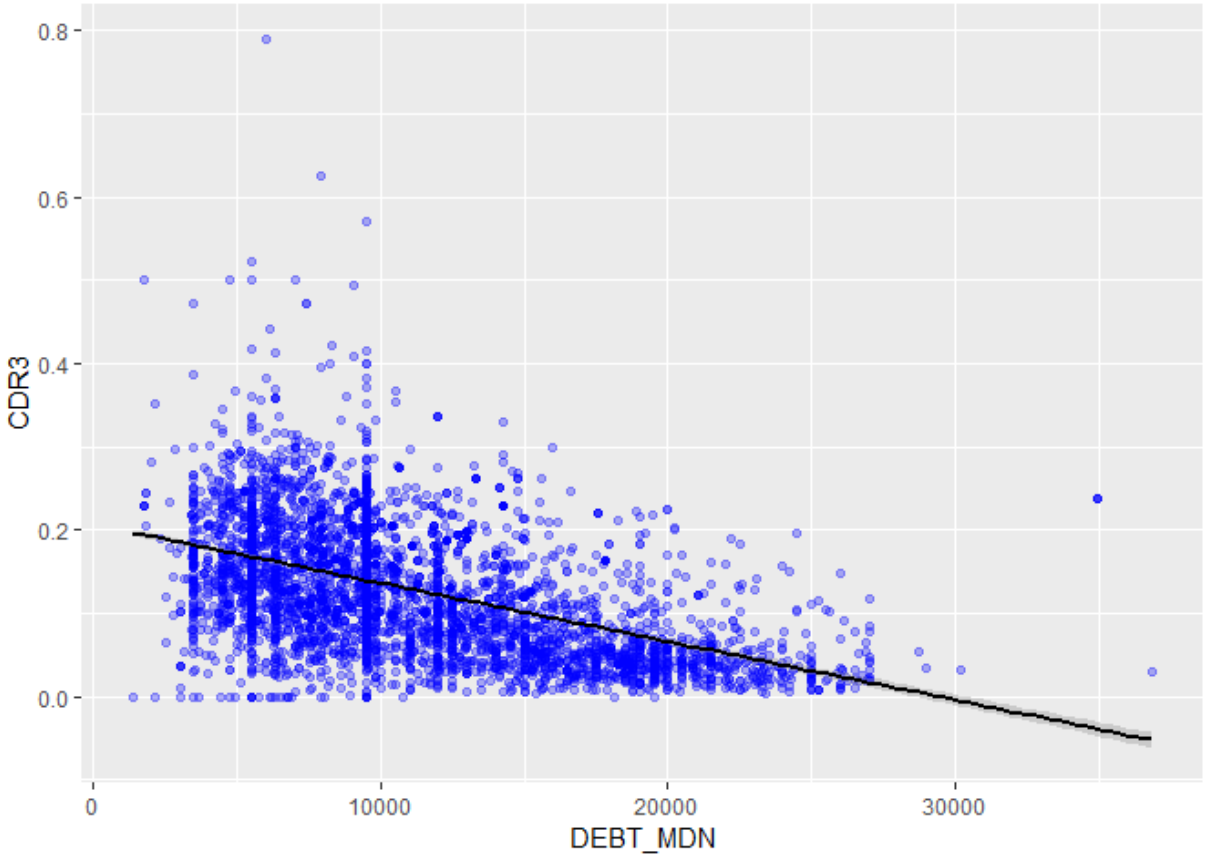
Average family income of independent students vs default rate

```
ggplot(sc1415.net,aes(x=IND_INC_AVG,y=CDR3)) +
   geom_point(alpha=.3, col='blue') +
   geom_smooth(method="lm", col="black")
```



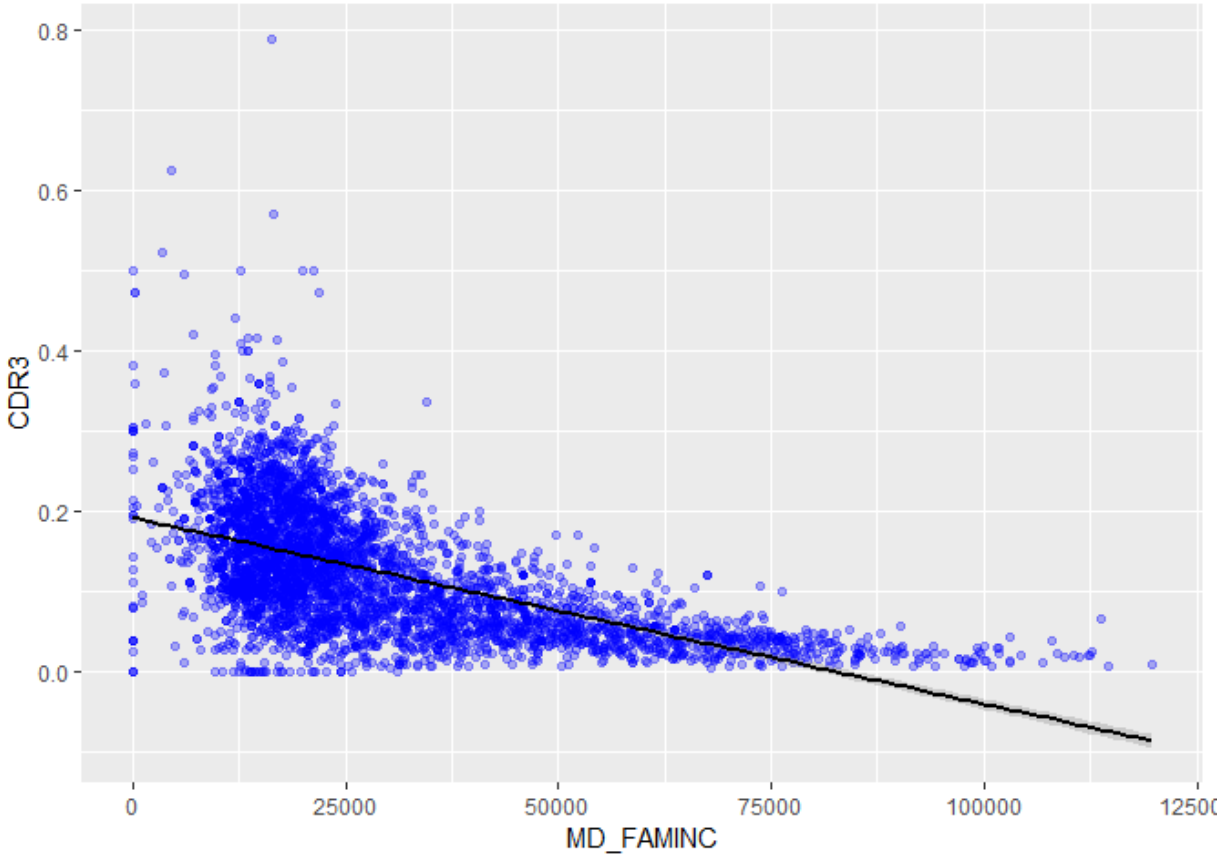
Median loan amount vs default rate

```
ggplot(sc1415.net,aes(x=DEBT_MDN,y=CDR3)) +
   geom_point(alpha=.3, col='blue') +
   geom_smooth(method="lm", col="black")
```



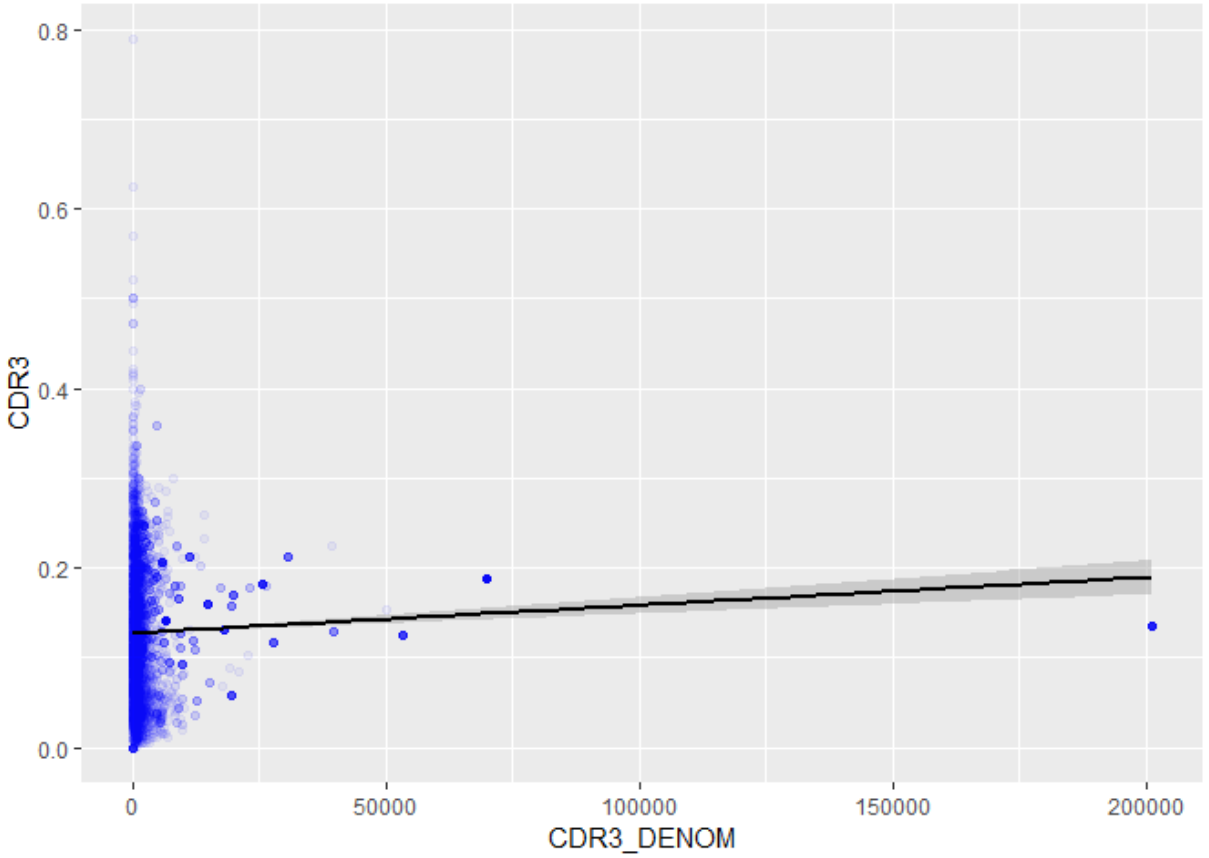
Median family income vs default rate

```
ggplot(sc1415.net,aes(x=MD_FAMINC,y=CDR3)) +
   geom_point(alpha=.3, col='blue') +
   geom_smooth(method="lm", col="black")
```



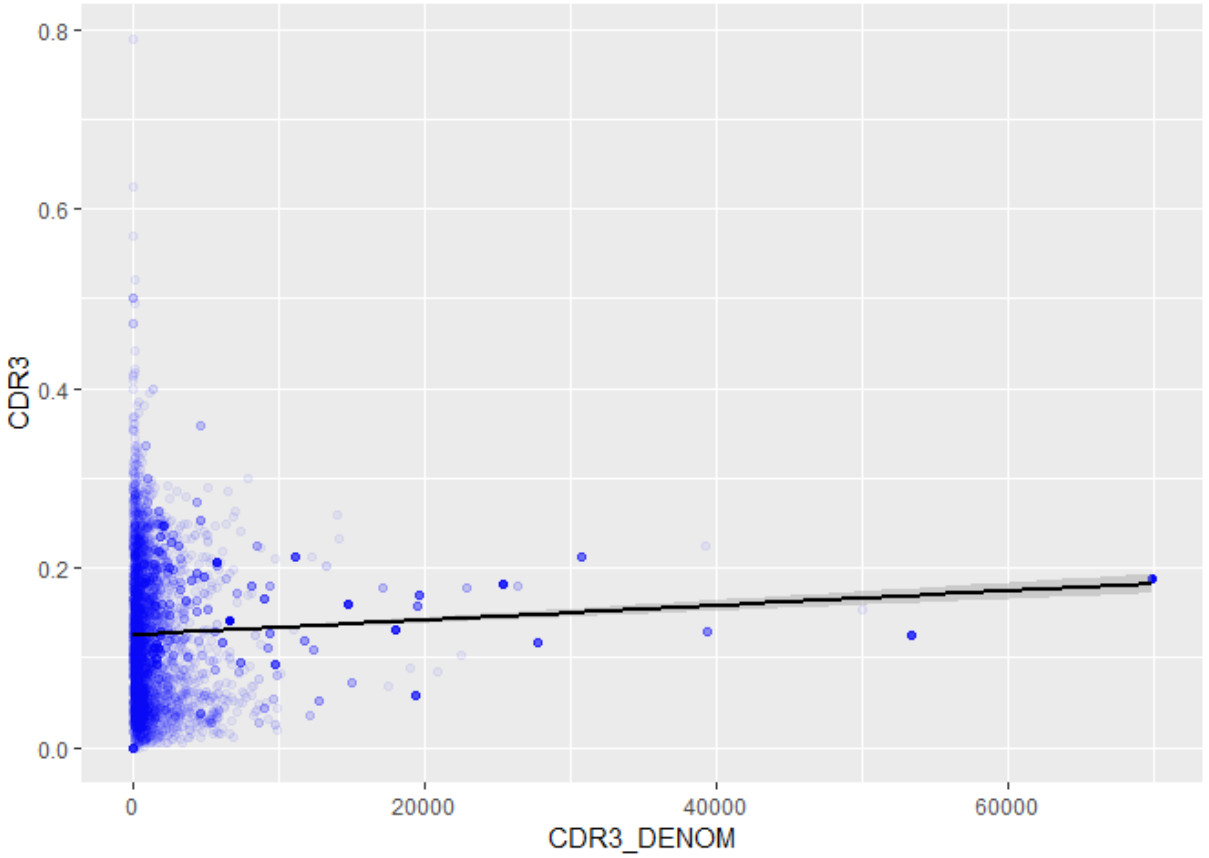
Number of students in cohort vs default rate

```
ggplot(sc1415.net,aes(x=CDR3_DENOM,y=CDR3)) +
   geom_point(alpha=.05, col='blue') +
   geom_smooth(method="lm", col="black")
```



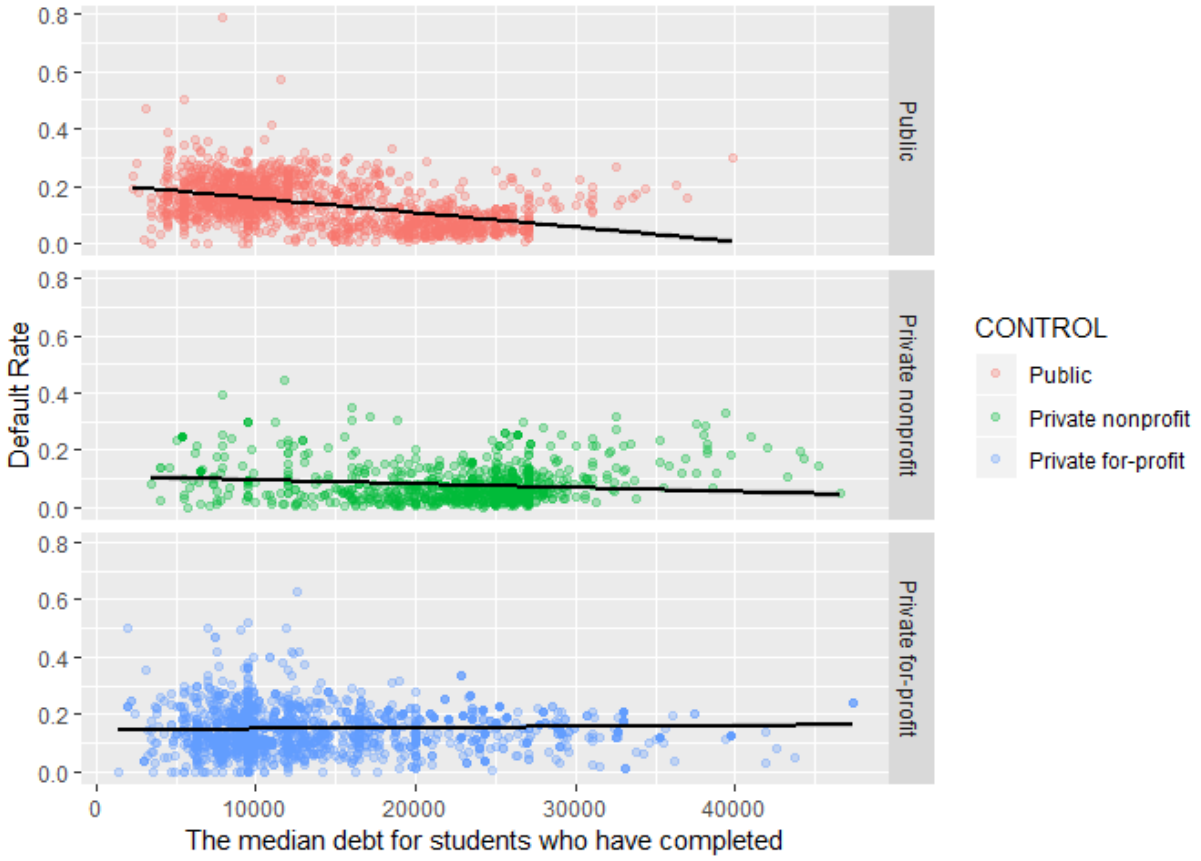
Number of students in cohort vs default rate - without outliers

```
ggplot(sc1415.net[sc1415.net$CDR3_DENOM<100000,],aes(x=CDR3_DENOM,y=CDR3)) +
   geom_point(alpha=.05, col='blue') +
   geom_smooth(method="lm", col="black")
```



Median debt vs default rate for students who completed by school ownership type

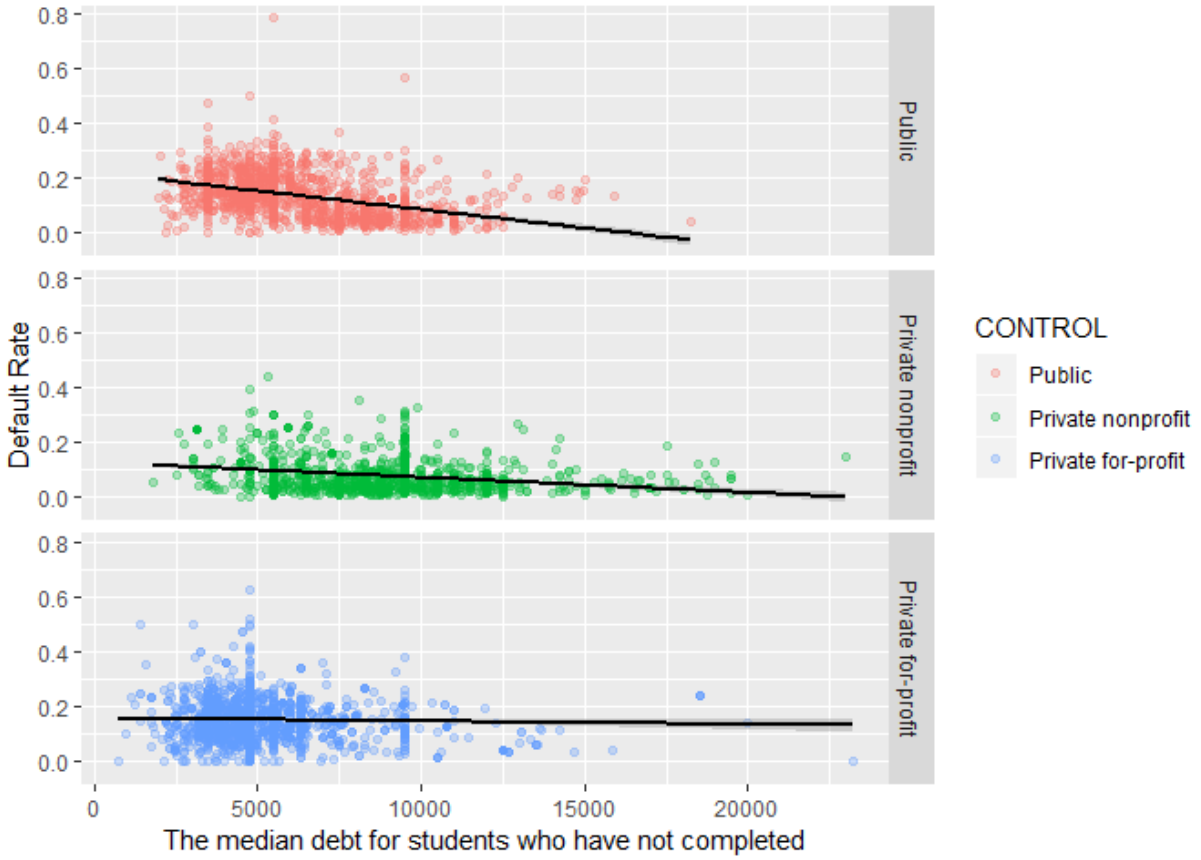
```
# The median debt for students who have completed vs default rate by school ownership

ggplot(sc1415.net,aes(x=GRAD_DEBT_MDN,y=CDR3, col=CONTROL)) +
   geom_point(alpha=.3) +
   facet_grid(CONTROL~.) +
   geom_smooth(col="black", method="lm") +
   xlab("The median debt for students who have completed") +
   ylab("Default Rate")
```



For public schools, the higher default rates are more concentrated in the lower median debt brackets. It is interesting to note that median debt has no association with default rate for for-profit private schools.

Median debt vs default rate for students who did not complete by school ownership type

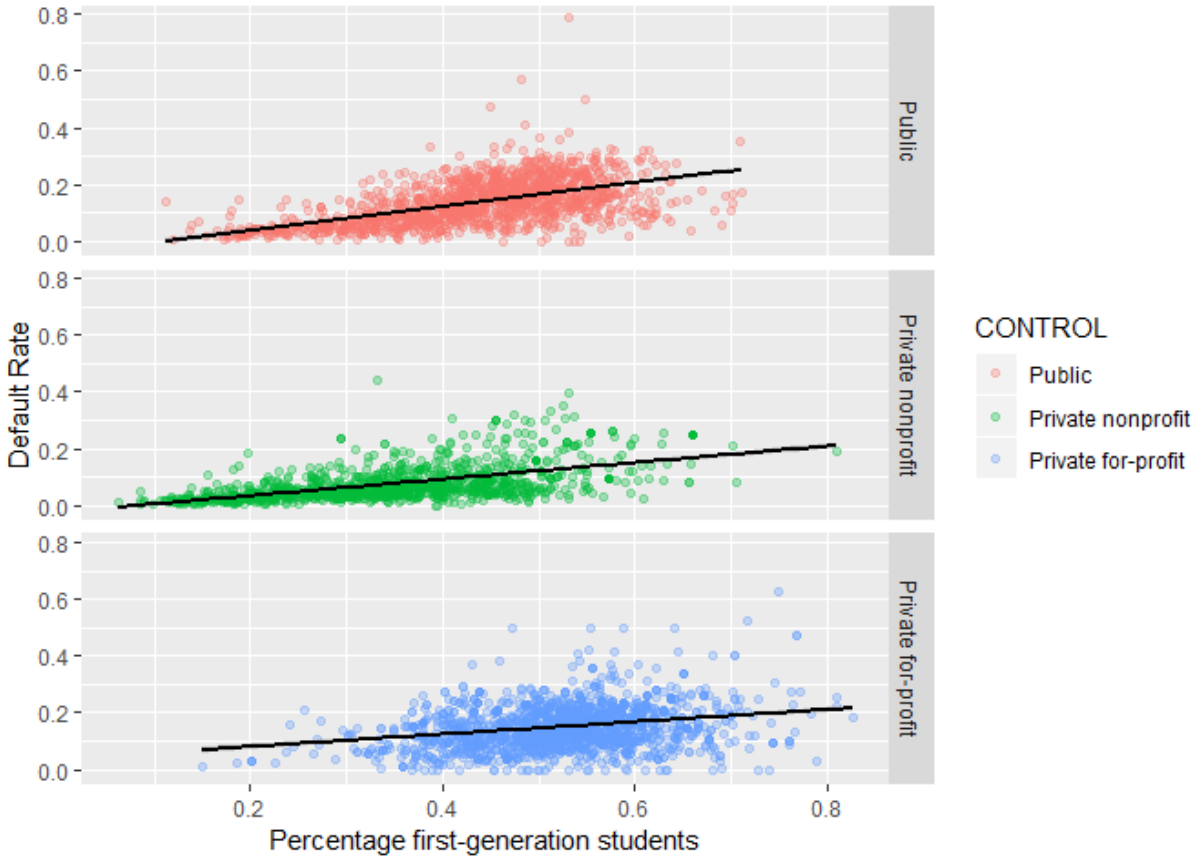
```
# The median debt for students who have withdrawn vs default rate by school ownership

ggplot(sc1415.net,aes(y=CDR3,x=WDRAW_DEBT_MDN, col=CONTROL)) +
   geom_point(alpha=.3) +
   facet_grid(CONTROL~.) +
   geom_smooth(col="black", method="lm") +
   xlab("The median debt for students who have not completed") +
   ylab("Default Rate")
```



The same trends are observed regardless of program completion status.

First generation vs default rate by school ownership type

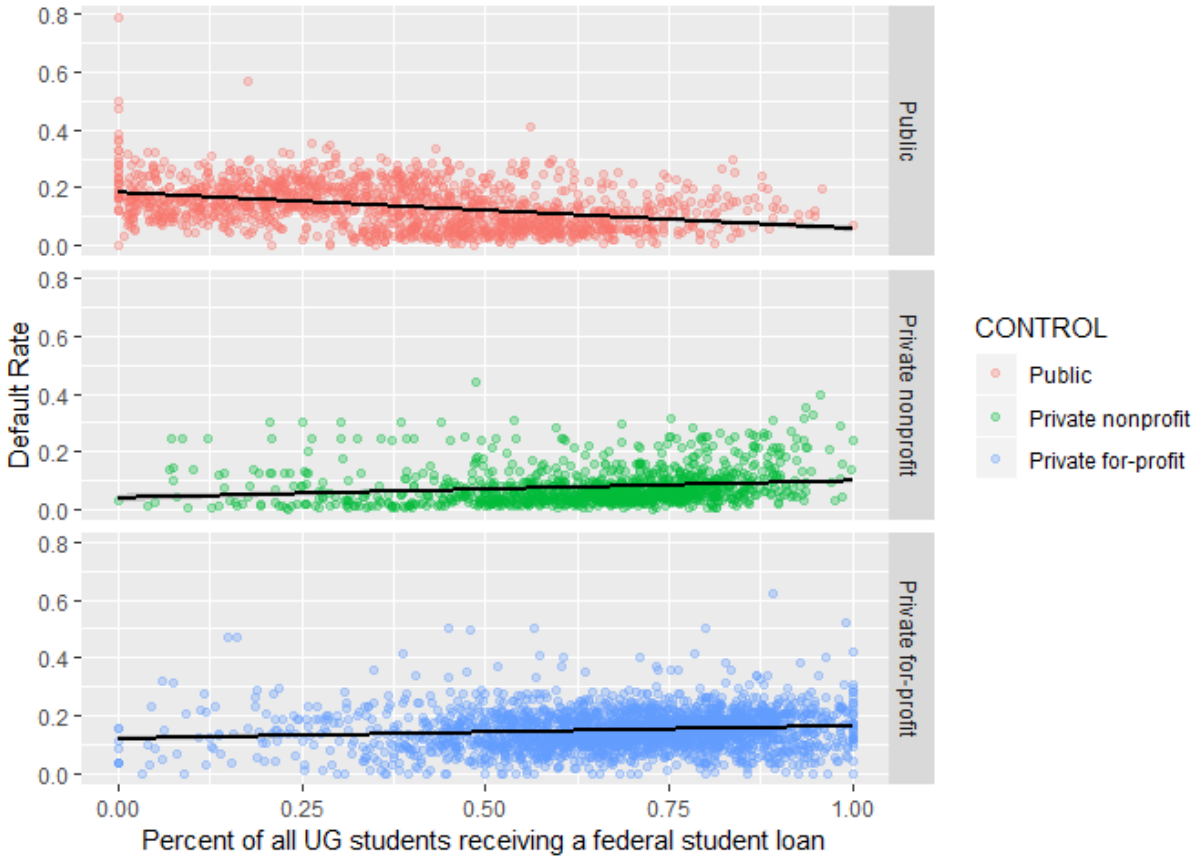
```
# Percentage first-generation students vs default rate by school ownership

ggplot(sc1415.net,aes(y=CDR3,x=PAR_ED_PCT_1STGEN, col=CONTROL)) +
   geom_point(alpha=.3) +
   facet_grid(CONTROL~.) +
   geom_smooth(col="black", method="lm") +
   xlab("Percentage first-generation students") +
   ylab("Default Rate")
```



There is a positive correlation between default rate and proportion of students who reported as first generation in getting higher education. For-profit private schools seem to have a higher mean of the proportions.

Federal student oan vs default rate by school ownership type

```
# Percent of all undergraduate students receiving a federal student loan

ggplot(sc1415.net,aes(y=CDR3,x=PCTFLOAN, col=CONTROL)) +
   geom_point(alpha=.3) +
   facet_grid(CONTROL~.) +
   geom_smooth(col="black", method="lm") +
   xlab("Percent of all UG students receiving a federal student loan") +
   ylab("Default Rate")
```

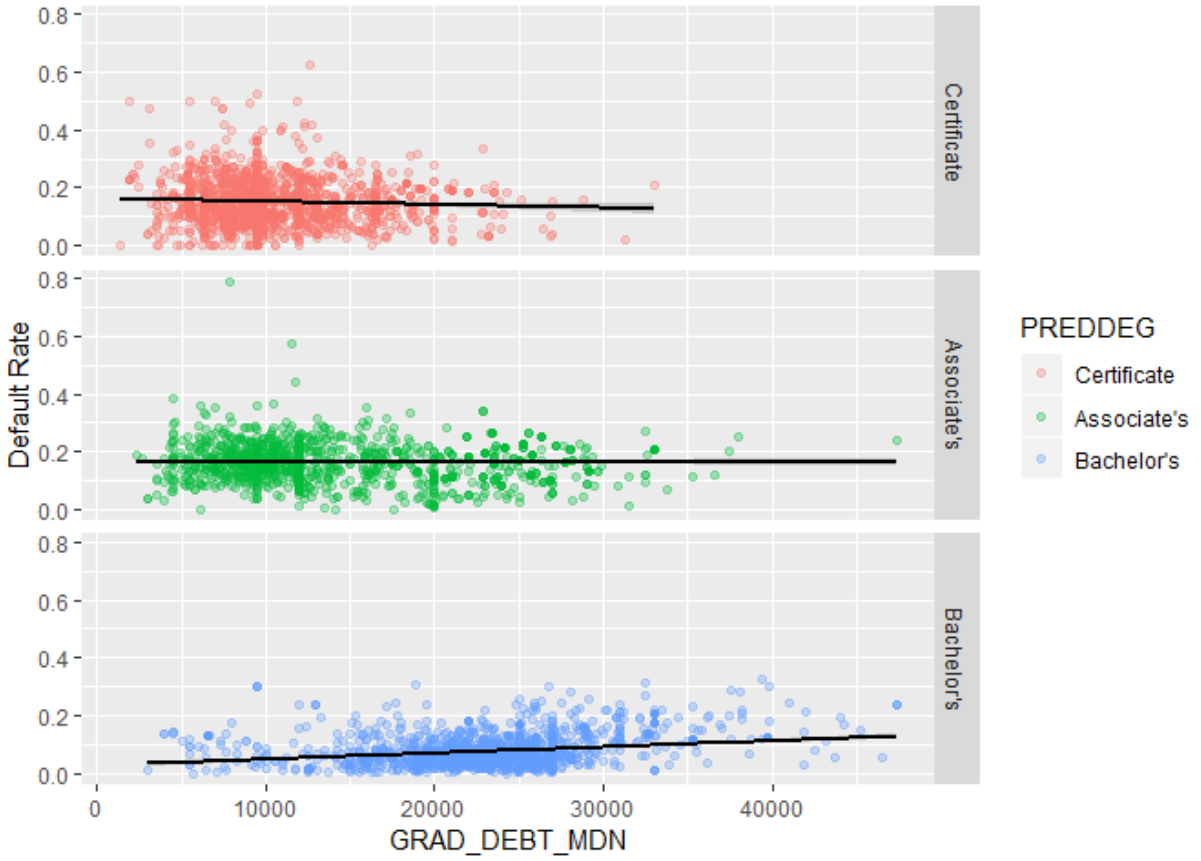


For-profit private institutions report higher fed loan participation rates. Yet, their default rates don't have any correlation with the participation rates.

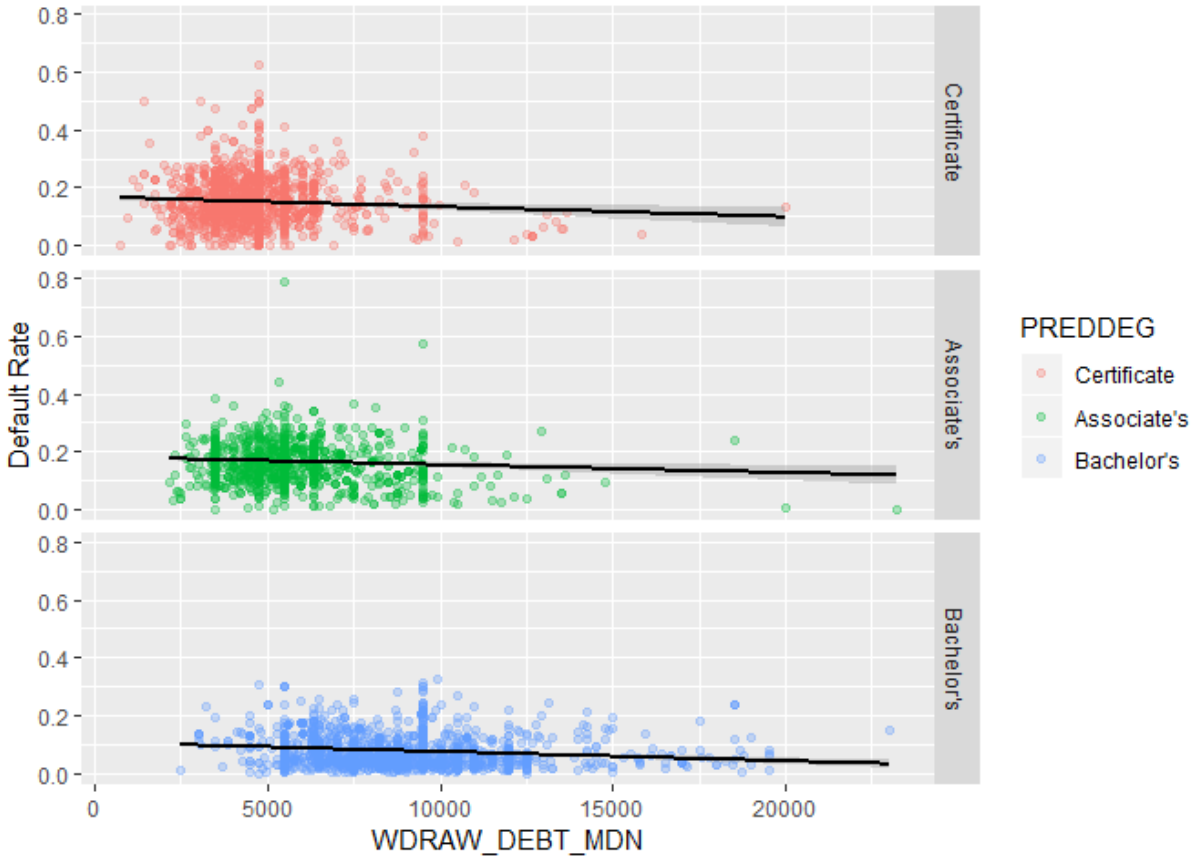
In the following plots facetting based on degree type or region, the correlation between first generation and default rate repeats. Students who completed bachelor's degree programs contributed to higher default rates when facing higher loan debt (positive correlation).

Across regions, simiar regression lines are drawn. No regions do better or worse when it comes to default rates.

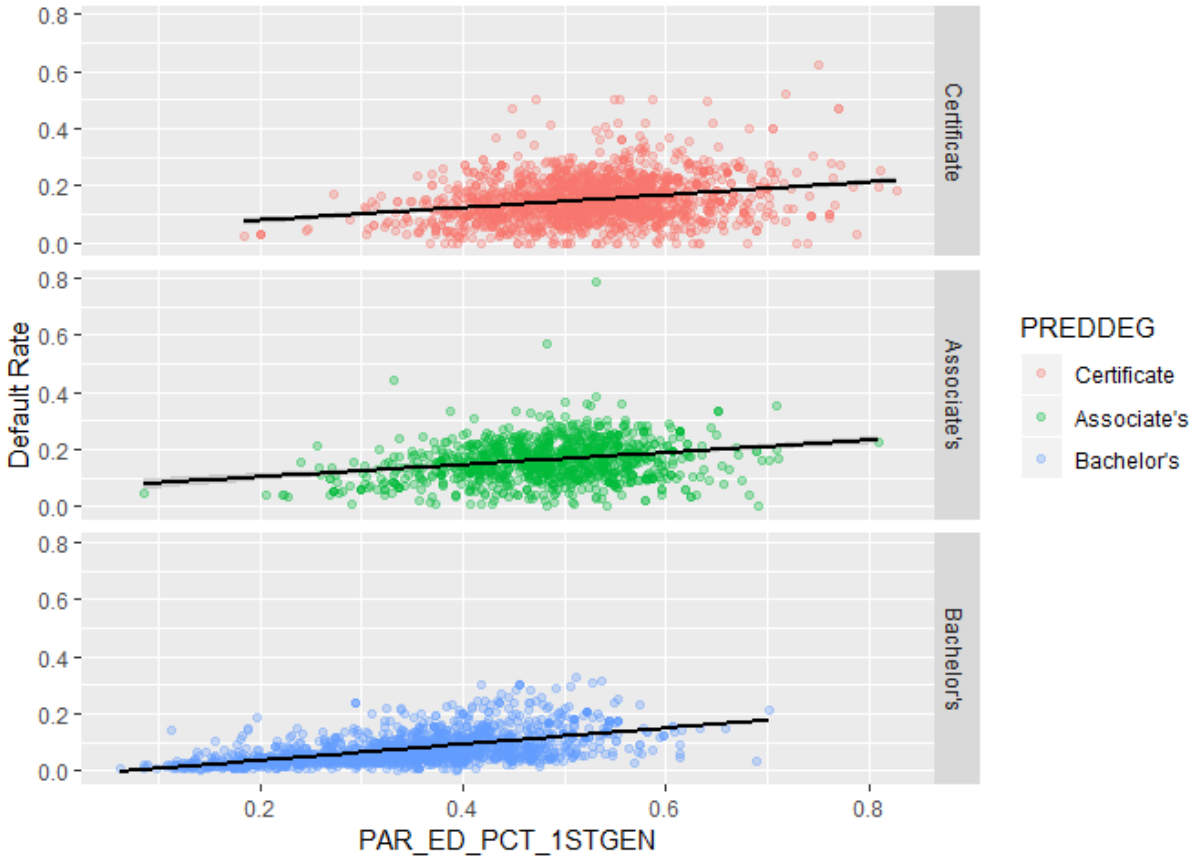
```
# The median debt for students who have completed vs default rate by degree type

ggplot(sc1415.net,aes(y=CDR3,x=GRAD_DEBT_MDN, col=PREDDEG)) +
   geom_point(alpha=.3) +
   facet_grid(PREDDEG~.) +
   geom_smooth(col="black", method="lm") +
   ylab("Default Rate")
```




```
# The median debt for students who have withdrawn vs default rate by degree type

ggplot(sc1415.net,aes(y=CDR3,x=WDRAW_DEBT_MDN, col=PREDDEG)) +
   geom_point(alpha=.3) +
   facet_grid(PREDDEG~.) +
   geom_smooth(col="black", method="lm") +
   ylab("Default Rate")
```

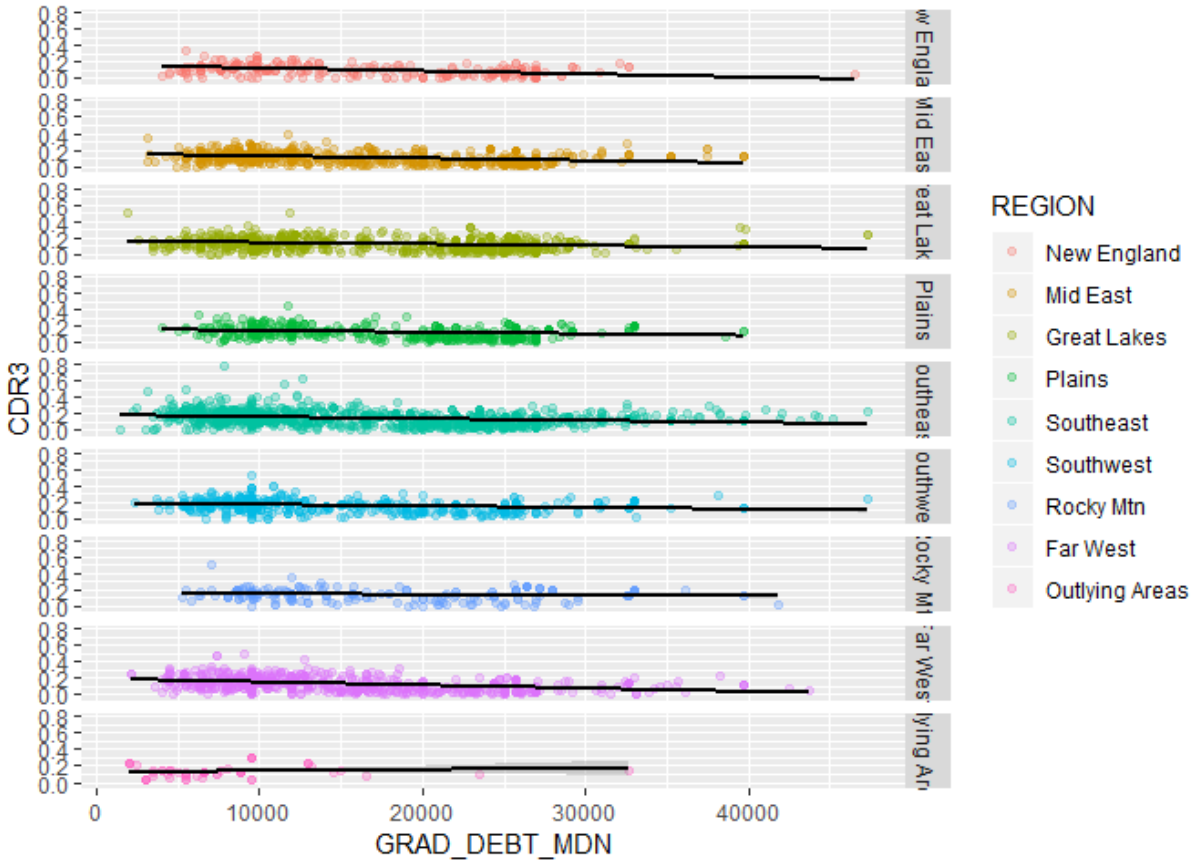



```
# Percentage first-generation students vs default rate by degree type

# try geom_jitter()


ggplot(sc1415.net,aes(y=CDR3,x=PAR_ED_PCT_1STGEN, col=PREDDEG)) +
   geom_point(alpha=.3) +
   facet_grid(PREDDEG~.) +
   geom_smooth(col="black", method="lm") +
   ylab("Default Rate")
```

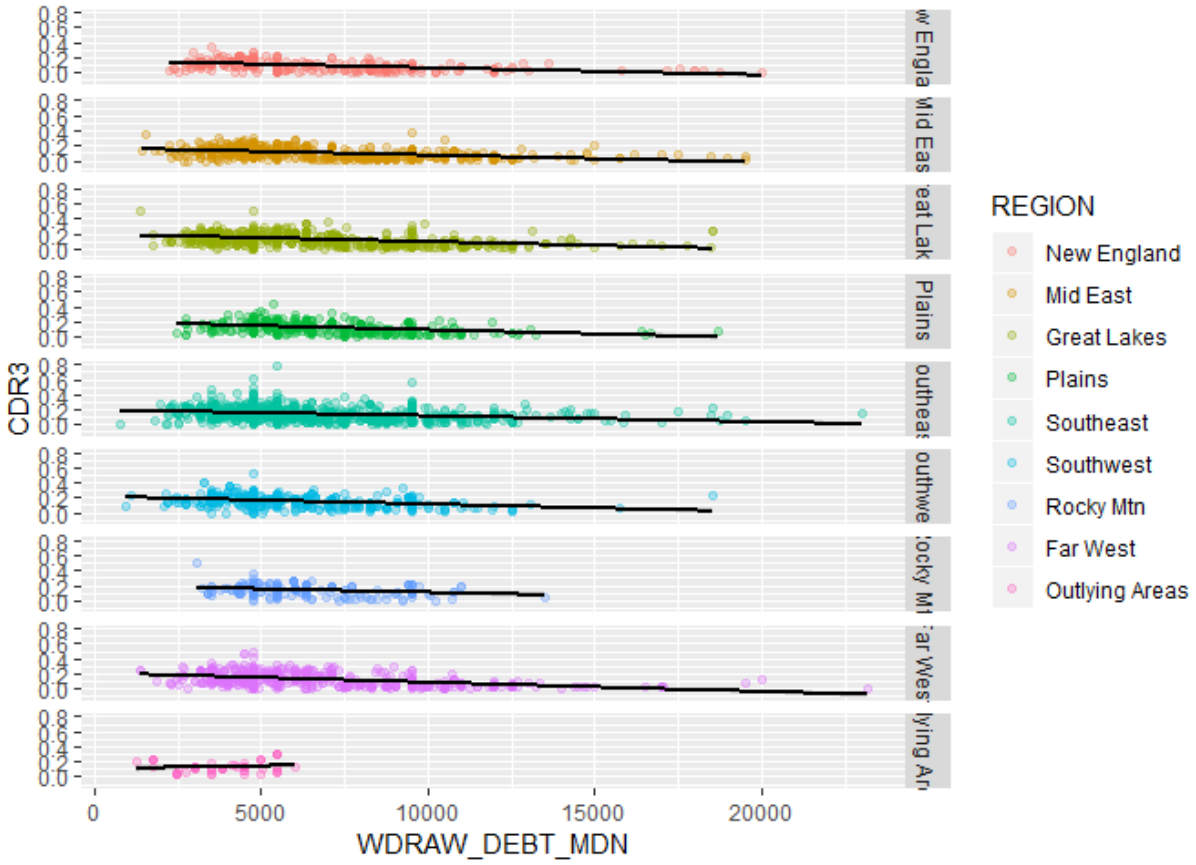


Numeric variables vs default rate by region.

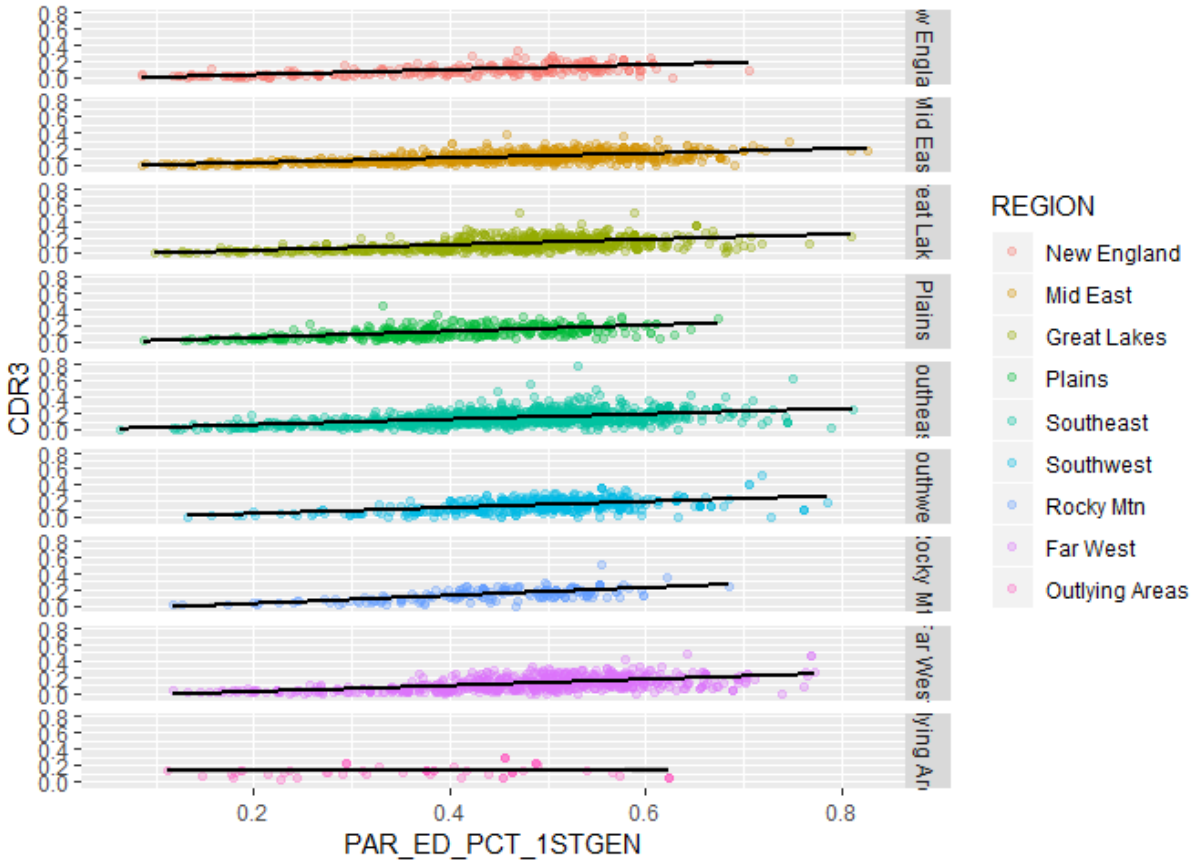
```
# The median debt for students who have completed vs default rate by region

ggplot(sc1415.net,aes(y=CDR3,x=GRAD_DEBT_MDN, col=REGION)) +
   geom_point(alpha=.3) +
   facet_grid(REGION~.) +
   geom_smooth(col="black", method="lm")
```




```
# The median debt for students who have withdrawn vs default rate by region

ggplot(sc1415.net,aes(y=CDR3,x=WDRAW_DEBT_MDN, col=REGION)) +
   geom_point(alpha=.3) +
   facet_grid(REGION~.) +
   geom_smooth(col="black", method="lm")
```




```
# Percentage first-generation students vs default rate by region

# try geom_jitter()


ggplot(sc1415.net,aes(y=CDR3,x=PAR_ED_PCT_1STGEN, col=REGION)) +
   geom_point(alpha=.3) +
   facet_grid(REGION~.) +
   geom_smooth(col="black", method="lm")
```



Categorical Variable Conversion to Dummy Variables

Before building modeling, categorical variables need to be converted to dummy variables. `CONTROL` , `PREDDEG` , and `REGION` will be converted. `DISTANCEONLY` has been dropped as its values are highly skewed.

```
library(caret)
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:survival':
##
##     cluster
```

```
dummies <- dummyVars("~ CONTROL + PREDDEG + REGION", data=sc1415.net,fullRank=TRUE)
dummies <- data.frame(predict(dummies,newdata=sc1415.net))
sc1415.final <- as.data.frame(cbind(sc1415.net,dummies))

# remove variables unused in modeling building

sc1415.final$OPEID6 <-  NULL
sc1415.final$STABBR <-  NULL
sc1415.final$INSTNM <-  NULL

sc1415.final$CONTROL <- NULL
sc1415.final$REGION  <- NULL
sc1415.final$PREDDEG <- NULL
sc1415.final$DISTANCEONLY <- NULL
```

Predictive Model Building

Now, the `sc1415.final` data frame is all set for building models. It consists of 5210 observations and 27 independent variables. The `CDR3` is the outcome variable. The goal is to predict student loan default rates.

Let's split the data frame to training and test sets.

```
library(caTools)
set.seed(100)
split_vec <- sample.split(sc1415.final$CDR3,SplitRatio=.75)
Train <- sc1415.final[split_vec,]
Test <- sc1415.final[!(split_vec),]
```

Model 1 - Linear Regression

Full MOdel

In this modeling, all independent variables are thrown in. Subsequently, uninfluential variables are eliminated with the `stepAIC()` function from the `MASS` package.

```
lm1 <- lm(CDR3~.,data=Train)
summary(lm1)
```

```
##
## Call:
## lm(formula = CDR3 ~ ., data = Train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20208 -0.03155 -0.00326  0.02692  0.59081
##
## Coefficients: (1 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.535e-01  1.363e-02  11.261  < 2e-16 ***
## NUMBRANCH                    1.088e-05  4.819e-05   0.226 0.821406
## TUITFTE                      1.494e-07  1.973e-07   0.757 0.448988
## INEXPFTE                    -8.360e-07  1.795e-07  -4.658 3.29e-06 ***
## PCTPELL                      2.325e-02  9.625e-03   2.416 0.015732 *
## PCTFLOAN                    -7.679e-03  7.737e-03  -0.992 0.321033
## PAR_ED_PCT_1STGEN            7.680e-02  1.621e-02   4.738 2.23e-06 ***
## DEP_INC_AVG                 -4.903e-07  1.811e-07  -2.707 0.006822 **
## IND_INC_AVG                 -1.792e-06  1.514e-07 -11.842  < 2e-16 ***
## DEBT_MDN                    -3.027e-06  4.409e-07  -6.866 7.65e-12 ***
## GRAD_DEBT_MDN                2.241e-06  2.647e-07   8.468  < 2e-16 ***
## WDRAW_DEBT_MDN               1.518e-06  6.433e-07   2.360 0.018303 *
## FAMINC                      -6.297e-07  3.750e-07  -1.679 0.093191 .
## MD_FAMINC                    4.389e-07  3.018e-07   1.454 0.145999
## CDR3_DENOM                   2.412e-08  5.735e-08   0.421 0.674054
## CONTROL.Private.nonprofit   -9.530e-03  3.313e-03  -2.876 0.004045 **
## CONTROL.Private.for.profit -2.765e-02  3.705e-03  -7.462 1.04e-13 ***
## PREDDEG.Associate.s          2.781e-03  3.070e-03   0.906 0.365057
## PREDDEG.Bachelor.s          -4.488e-02  4.481e-03 -10.015  < 2e-16 ***
## REGION.New.England           2.280e-02  1.088e-02   2.096 0.036133 *
## REGION.Mid.East              1.147e-02  1.038e-02   1.105 0.269339
## REGION.Great.Lakes           1.904e-02  1.033e-02   1.843 0.065408 .
## REGION.Plains                3.162e-02  1.045e-02   3.024 0.002510 **
## REGION.Southeast             3.063e-02  9.950e-03   3.079 0.002094 **
## REGION.Southwest             3.921e-02  1.027e-02   3.817 0.000137 ***
## REGION.Rocky.Mtn             3.820e-02  1.101e-02   3.469 0.000527 ***
## REGION.Far.West              1.884e-02  1.022e-02   1.843 0.065389 .
## REGION.Outlying.Areas               NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05592 on 3897 degrees of freedom
## Multiple R-squared:  0.4677, Adjusted R-squared:  0.4642
## F-statistic: 131.7 on 26 and 3897 DF,  p-value: < 2.2e-16
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:plotly':
##
##     select
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
step.model1 <- stepAIC(lm1, direction = "both",
                       trace = FALSE)
summary(step.model1)
```

```
##
## Call:
## lm(formula = CDR3 ~ INEXPFTE + PCTPELL + PAR_ED_PCT_1STGEN +
##     DEP_INC_AVG + IND_INC_AVG + DEBT_MDN + GRAD_DEBT_MDN + WDRAW_DEBT_MDN +
##     CONTROL.Private.nonprofit + CONTROL.Private.for.profit +
##     PREDDEG.Bachelor.s + REGION.New.England + REGION.Great.Lakes +
##     REGION.Plains + REGION.Southeast + REGION.Southwest + REGION.Rocky.Mtn +
##     REGION.Far.West, data = Train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20264 -0.03203 -0.00332  0.02691  0.59507
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.616e-01  1.068e-02  15.138  < 2e-16 ***
## INEXPFTE                    -7.903e-07  1.605e-07  -4.923 8.89e-07 ***
## PCTPELL                      1.640e-02  6.635e-03   2.472 0.013477 *
## PAR_ED_PCT_1STGEN            8.273e-02  1.483e-02   5.580 2.57e-08 ***
## DEP_INC_AVG                 -7.232e-07  9.468e-08  -7.639 2.74e-14 ***
## IND_INC_AVG                 -1.745e-06  1.450e-07 -12.039  < 2e-16 ***
## DEBT_MDN                    -3.169e-06  4.046e-07  -7.833 6.11e-15 ***
## GRAD_DEBT_MDN                2.461e-06  2.062e-07  11.937  < 2e-16 ***
## WDRAW_DEBT_MDN               1.495e-06  6.213e-07   2.405 0.016200 *
## CONTROL.Private.nonprofit   -1.007e-02  2.960e-03  -3.403 0.000674 ***
## CONTROL.Private.for.profit -2.824e-02  2.771e-03 -10.192  < 2e-16 ***
## PREDDEG.Bachelor.s          -4.939e-02  3.214e-03 -15.367  < 2e-16 ***
## REGION.New.England           1.181e-02  4.380e-03   2.697 0.007025 **
## REGION.Great.Lakes           8.262e-03  3.165e-03   2.611 0.009064 **
## REGION.Plains                2.121e-02  3.777e-03   5.615 2.10e-08 ***
## REGION.Southeast             1.976e-02  2.906e-03   6.801 1.19e-11 ***
## REGION.Southwest             2.831e-02  3.586e-03   7.895 3.74e-15 ***
## REGION.Rocky.Mtn             2.797e-02  5.169e-03   5.411 6.65e-08 ***
## REGION.Far.West              8.469e-03  3.347e-03   2.530 0.011444 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05591 on 3905 degrees of freedom
## Multiple R-squared:  0.4668, Adjusted R-squared:  0.4644
## F-statistic: 189.9 on 18 and 3905 DF,  p-value: < 2.2e-16
```

It reduces the number of predictors to 19 from 28.

Alternatively, let's try the `regsubsets()` that provides the tuning parameter `nvmax` for the number of predictors. In order to optimize `nvmax` , we can use k-fold cross-validation. Click here (http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/154-stepwise-regression-essentials-in-r/) for details and an example.

```
set.seed(100)

k = 10
train.control <- trainControl(method = "cv", number = k)

nvmaxCV <- train(CDR3 ~., data = sc1415.final,
                    method = "leapBackward",
                    tuneGrid = data.frame(nvmax = 1:19),
                    trControl = train.control
                    )
nvmaxCV$results
```

```
##    nvmax       RMSE  Rsquared        MAE      RMSESD RsquaredSD
## 1      1 0.05880183 0.3820780 0.04402567 0.003158120 0.03002598
## 2      2 0.05755215 0.4079943 0.04239540 0.003191844 0.03039607
## 3      3 0.05712145 0.4169581 0.04152682 0.003342298 0.03351274
## 4      4 0.05640868 0.4315194 0.04087356 0.003537919 0.03618543
## 5      5 0.05586732 0.4425443 0.04061163 0.003520209 0.03654763
## 6      6 0.05549743 0.4500348 0.04034471 0.003465128 0.03775071
## 7      7 0.05525619 0.4547554 0.04010286 0.003384135 0.03619779
## 8      8 0.05520880 0.4557831 0.04002321 0.003325895 0.03475417
## 9      9 0.05505228 0.4588362 0.03987312 0.003326339 0.03499265
## 10    10 0.05495476 0.4607917 0.03982846 0.003310868 0.03459318
## 11    11 0.05487428 0.4623041 0.03980066 0.003258281 0.03375367
## 12    12 0.05474083 0.4649687 0.03969146 0.003189000 0.03270093
## 13    13 0.05447665 0.4700986 0.03957972 0.003238883 0.03361240
## 14    14 0.05455456 0.4685896 0.03961175 0.003333075 0.03480664
## 15    15 0.05457591 0.4681576 0.03963830 0.003314591 0.03446435
## 16    16 0.05457008 0.4682438 0.03965770 0.003325774 0.03451568
## 17    17 0.05449215 0.4697857 0.03957562 0.003335199 0.03481860
## 18    18 0.05449837 0.4696842 0.03952565 0.003339718 0.03508488
## 19    19 0.05448980 0.4698766 0.03950481 0.003336922 0.03498536
##          MAESD
## 1  0.001525026
## 2  0.001395055
## 3  0.001537490
## 4  0.001716406
## 5  0.001877639
## 6  0.001788714
## 7  0.001772634
## 8  0.001766278
## 9  0.001791227
## 10 0.001839421
## 11 0.001817560
## 12 0.001755109
## 13 0.001791195
## 14 0.001829495
## 15 0.001806506
## 16 0.001819638
## 17 0.001836434
## 18 0.001853320
## 19 0.001830632
```

The # of predictors recommended is 13. The list of the selected predictors is as follows:

```
names(coef(nvmaxCV$finalModel,13))
```

```
##  [1] "(Intercept)"                "INEXPFTE"
##  [3] "PAR_ED_PCT_1STGEN"          "DEP_INC_AVG"
##  [5] "IND_INC_AVG"                "DEBT_MDN"
##  [7] "GRAD_DEBT_MDN"              "CONTROL.Private.nonprofit"
##  [9] "CONTROL.Private.for.profit" "PREDDEG.Bachelor.s"
## [11] "REGION.Plains"              "REGION.Southeast"
## [13] "REGION.Southwest"           "REGION.Rocky.Mtn"
```

Let's build another model using these variables.

```
lm2 <- lm(CDR3~INEXPFTE+PAR_ED_PCT_1STGEN+DEP_INC_AVG+IND_INC_AVG+
          DEBT_MDN+GRAD_DEBT_MDN+CONTROL.Private.nonprofit+CONTROL.Private.for.profit+
          PREDDEG.Bachelor.s+REGION.Plains+REGION.Southeast+REGION.Southwest+REGION.Rocky.Mtn,
          data=Train)

summary(lm2)
```

```
##
## Call:
## lm(formula = CDR3 ~ INEXPFTE + PAR_ED_PCT_1STGEN + DEP_INC_AVG +
##     IND_INC_AVG + DEBT_MDN + GRAD_DEBT_MDN + CONTROL.Private.nonprofit +
##     CONTROL.Private.for.profit + PREDDEG.Bachelor.s + REGION.Plains +
##     REGION.Southeast + REGION.Southwest + REGION.Rocky.Mtn, data = Train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20340 -0.03213 -0.00382  0.02702  0.59422
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 1.794e-01  9.754e-03  18.397  < 2e-16 ***
## INEXPFTE                   -7.677e-07  1.605e-07  -4.784 1.78e-06 ***
## PAR_ED_PCT_1STGEN           8.609e-02  1.475e-02   5.835 5.81e-09 ***
## DEP_INC_AVG                -8.273e-07  8.751e-08  -9.454  < 2e-16 ***
## IND_INC_AVG                -1.722e-06  1.423e-07 -12.101  < 2e-16 ***
## DEBT_MDN                   -2.655e-06  3.603e-07  -7.369 2.08e-13 ***
## GRAD_DEBT_MDN               2.671e-06  1.968e-07  13.578  < 2e-16 ***
## CONTROL.Private.nonprofit  -9.639e-03  2.909e-03  -3.313  0.00093 ***
## CONTROL.Private.for.profit -2.755e-02  2.492e-03 -11.056  < 2e-16 ***
## PREDDEG.Bachelor.s         -4.995e-02  3.193e-03 -15.642  < 2e-16 ***
## REGION.Plains               1.511e-02  3.305e-03   4.571 5.01e-06 ***
## REGION.Southeast            1.436e-02  2.255e-03   6.367 2.15e-10 ***
## REGION.Southwest            2.209e-02  3.078e-03   7.177 8.49e-13 ***
## REGION.Rocky.Mtn            2.183e-02  4.832e-03   4.518 6.44e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05604 on 3910 degrees of freedom
## Multiple R-squared:  0.4637, Adjusted R-squared:  0.462
## F-statistic: 260.1 on 13 and 3910 DF,  p-value: < 2.2e-16
```

The RMSEs from the three models above are:

```
# Full Model - 28 variables
sqrt(sum(lm1$residuals^2)/nrow(Train))
```

```
## [1] 0.05572979
```

```
# Model with 19 variables
sqrt(sum(step.model1$residuals^2)/nrow(Train))
```

```
## [1] 0.05577672
```

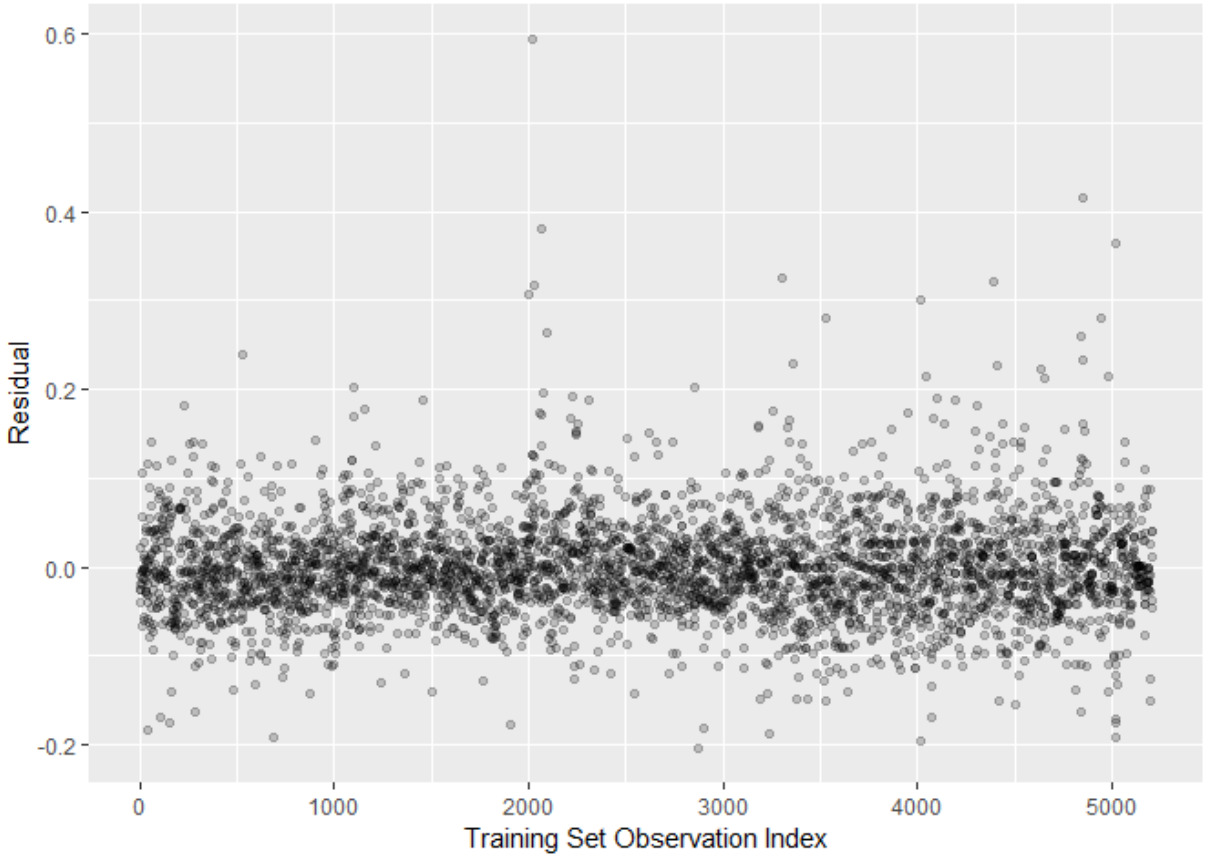
```
# Model with 13 variables
sqrt(sum(lm2$residuals^2)/nrow(Train))
```

```
## [1] 0.05593765
```

While the RMSE of the full model is the lowest of the three, the delta is quite small. Therefore, the `lm2` model will be used for prediction.

The following plot shows that the residuals bounce around y=0, confirming the validity of the model.

```
#plot(lm2$residuals)

ggplot(data=Train,aes(x=as.numeric(row.names(Train)),y=lm2$residuals)) + geom_point(alpha=.2) +
   xlab("Training Set Observation Index") +
   ylab("Residual")
```



```
# Prediction
lm2.pred <- predict(lm2, newdata=Test)
residualTest <- (lm2.pred - Test$CDR3)
# RMSE
sqrt(sum(residualTest^2)/nrow(Test))
```

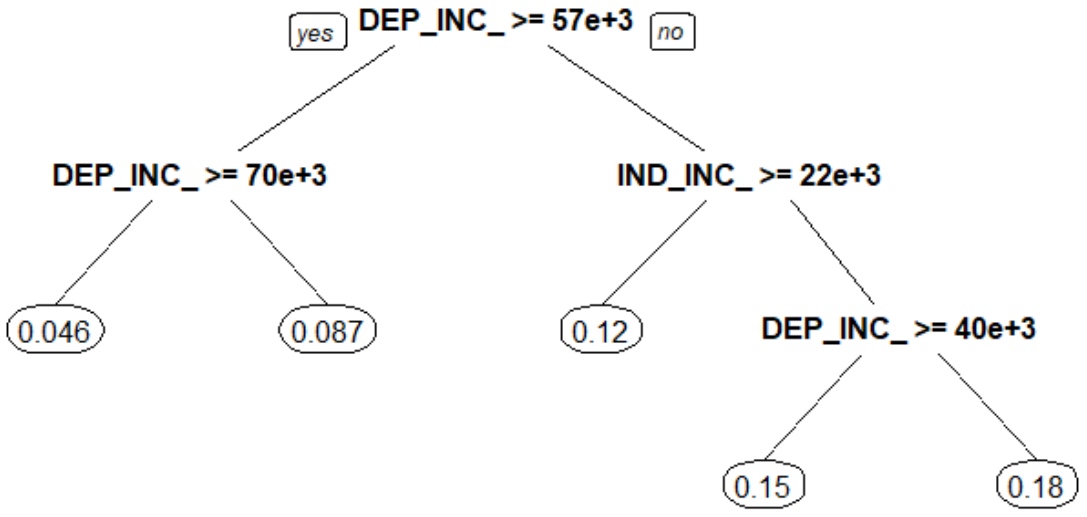
```
## [1] 0.04967096
```

The RMSE at 0.0497 is lower than that of the training set.

Model 2 - Classification and Regression Tree (CART)

In this modeling, we will create a decision tree whose end nodes of branches show average default rates. The 13 predictors identified in the linear regression model will be used.

```
library(rpart)
library(rpart.plot)

defaultsTree = rpart(CDR3 ~
                     INEXPFTE + PAR_ED_PCT_1STGEN + DEP_INC_AVG +
                     IND_INC_AVG + DEBT_MDN + GRAD_DEBT_MDN + CONTROL.Private.nonprofit +
                     CONTROL.Private.for.profit + PREDDEG.Bachelor.s + REGION.Plains +
                     REGION.Southeast + REGION.Southwest + REGION.Rocky.Mtn,
                     data=Train,
                     control=rpart.control(minbucket=4)  # sqrt of # of predicts in the model
)

prp(defaultsTree)  # plotting the tree
```



```
predictCART = predict(defaultsTree,newdata=Test)

residualTestCART <- (predictCART- Test$CDR3)
# RMSE
sqrt(sum(residualTestCART^2)/nrow(Test))
```

```
## [1] 0.05173243
```

The tree looks highly simple. It references only two variables for splitting – `DEP_INC_AVG' and` IND_INC_AVG`.

```
`DEP_INC_AVG`: Average family income of dependent students in real 2015 dollars.
`IND_INC_AVG`: Average family income of independent students in real 2015 dollars.
```

The RMSE is 0.0517

Model 3 - Random Forest (RF)

Random Forest lacks interpretability, but results in a better accuracy.

```
library("randomForest")
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

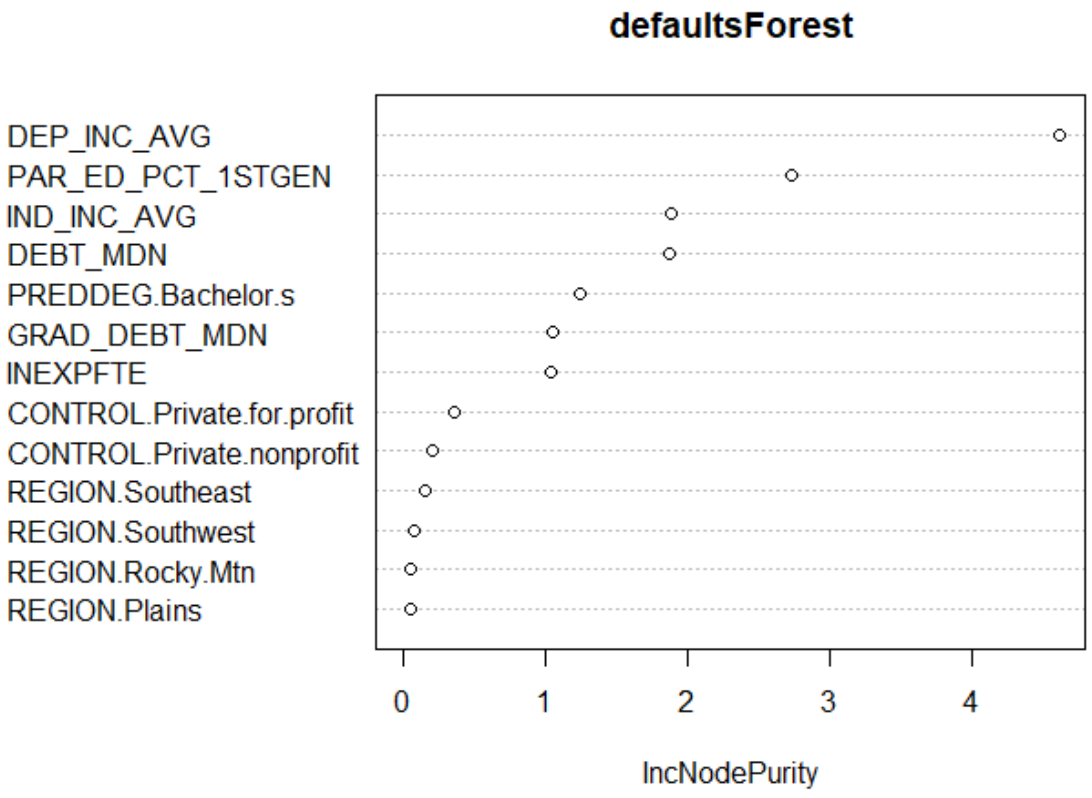
```
## The following object is masked from 'package:Hmisc':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
defaultsForest = randomForest(CDR3 ~
                     INEXPFTE + PAR_ED_PCT_1STGEN + DEP_INC_AVG +
                     IND_INC_AVG + DEBT_MDN + GRAD_DEBT_MDN + CONTROL.Private.nonprofit +
                     CONTROL.Private.for.profit + PREDDEG.Bachelor.s + REGION.Plains +
                     REGION.Southeast + REGION.Southwest + REGION.Rocky.Mtn,
                     data=Train,
                     nodesize=25,
                     ntree=200)
predictForest = predict(defaultsForest,newdata=Test)
residualTestForest <- (predictForest- Test$CDR3)

#RMSE
sqrt(sum(residualTestForest^2)/nrow(Test))
```

```
## [1] 0.0445705
```

```
varImpPlot(defaultsForest)
```



Its RMSE is indeed the lowest of the 3 models at 0.0446. The model identified the following seven as the top predictors:

`DEP_INC_AVG` : Average family income of dependent students in real 2015 dollars
`IND_INC_AVG` : Average family income of independent students in real 2015 dollars
`PAR_ED_PCT_1STGEN` : Percentage first-generation students
`DEBT_MDN` : The median original amount of the loan principal upon entering repayment
`PREDDEG.Bachelor.s` : Bachelor's being the predominant degree
`INEXPFTE` : Instructional expenditures per full-time equivalent student
`GRAD_DEBT_MDN` : The median debt for students who have completed