Student-Loan-Data

Camellia Debnath

11/28/2018

1. Combining the data for 2008-2013

```
col_09_10 <- read.csv("MERGED2009_10_PP.csv")
college_09_10 <- col_09_10 %>%
  mutate(GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         "Year" = "2009-10") %>%
   mutate(DEBT_TO_EARN = GRAD_DEBT_MDN/MD_EARN_WNE_P8)

col_11_12 <- read.csv("MERGED2011_12_PP.csv")
college_11_12 <- col_11_12 %>%
  mutate(GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         "Year" = "2011-12") %>%
   mutate(DEBT_TO_EARN = GRAD_DEBT_MDN/MD_EARN_WNE_P8)

col_12_13 <- read.csv("MERGED2012_13_PP.csv")
college_12_13 <- col_12_13 %>%
  mutate(GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         "Year" = "2012-13") %>%
   mutate(DEBT_TO_EARN = GRAD_DEBT_MDN/MD_EARN_WNE_P8)

col_13_14 <- read.csv("MERGED2013_14_PP.csv")
college_13_14 <- col_13_14 %>%
  mutate(GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         "Year" = "2013-14") %>%
   mutate(DEBT_TO_EARN = GRAD_DEBT_MDN/MD_EARN_WNE_P8)

col_14_15 <- read.csv("MERGED2014_15_PP.csv")
college_14_15 <- col_14_15 %>%
  mutate(GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         "Year" = "2014-15") %>%
   mutate(DEBT_TO_EARN = GRAD_DEBT_MDN/MD_EARN_WNE_P8)


college_09_15 <- rbind(college_09_10, college_11_12, college_12_13, college_13_14, college_14_15)
```

2. Partitioning the data into training and test sets

```
college_09_15_parts <- resample_partition(college_09_15 ,c(train = 0.6, valid = 0.2, test = 0.2))
```

```
college_09_15_train_ <- as_tibble(college_09_15_parts$train)
college_09_15_test_ <- as_tibble(college_09_15_parts$test)
college_09_15_valid_ <- as_tibble(college_09_15_parts$valid)
```

3. Subsetting variables to check for potential predictors

Intuitively, we select a subset of variables, and tidy the data for further EDA.

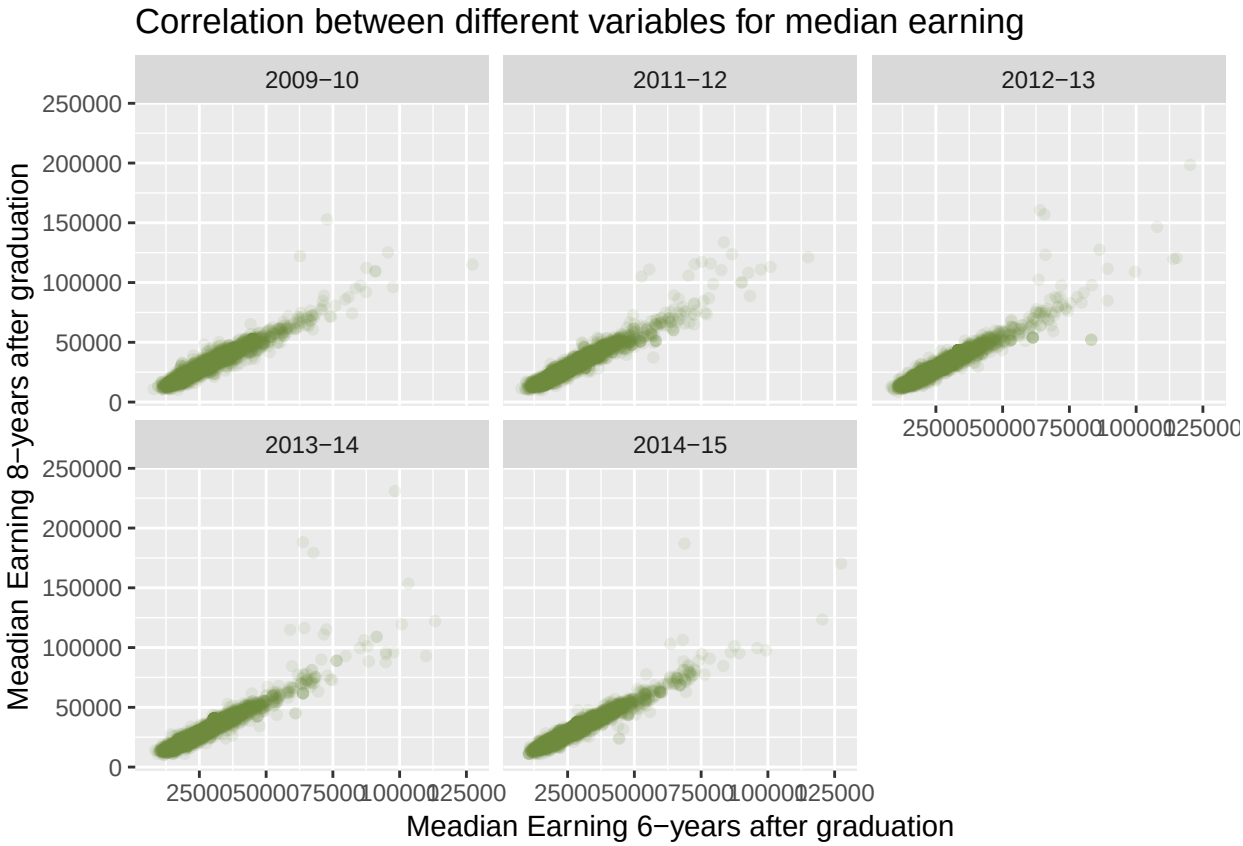
```
college_09_15_train <- college_09_15_train_ %>%
  select(INSTNM, COMPL_RPY_3YR_RT, GRAD_DEBT_MDN,PCTFLOAN, PCTPELL, MD_EARN_WNE_P6, MD_EARN_WNE_P8, MD_
         CDR3, MEDIAN_HH_INC, AGE_ENTRY, UGDS, CONTROL, COSTT4_A, COSTT4_P, Year, DEBT_TO_EARN, MD_FAMI
  mutate(COMPL_RPY_3YR_RT = as.numeric(as.character(COMPL_RPY_3YR_RT)),
         GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P6 = as.numeric(as.character(MD_EARN_WNE_P6)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         MD_EARN_WNE_P10 = as.numeric(as.character(MD_EARN_WNE_P10)),
         CDR3 = as.numeric(as.character(CDR3)), # three-year cohort default rate
         MEDIAN_HH_INC = as.numeric(as.character(MEDIAN_HH_INC)), #Median household income
         AGE_ENTRY = as.numeric(as.character(AGE_ENTRY)),
         UGDS = as.numeric(as.character(UGDS)),
         CONTROL =as.character(CONTROL),
         COSTT4_A = as.numeric(as.character(COSTT4_A)), #avg cost of attendance for academic year insti
         COSTT4_P = as.numeric(as.character(COSTT4_P)), #avg cost of attendance for program year instit
         PCTFLOAN = as.numeric(as.character(PCTFLOAN)),
         PCTPELL = as.numeric(as.character(PCTPELL)),
         MD_FAMINC = as.numeric(as.character(MD_FAMINC)))

college_09_15_test <- college_09_15_test_ %>%
  select(INSTNM, COMPL_RPY_3YR_RT, GRAD_DEBT_MDN,PCTFLOAN, PCTPELL, MD_EARN_WNE_P6, MD_EARN_WNE_P8, MD_
         CDR3, MEDIAN_HH_INC, AGE_ENTRY, UGDS, CONTROL, COSTT4_A, COSTT4_P, Year, DEBT_TO_EARN, MD_FAMI
  mutate(COMPL_RPY_3YR_RT = as.numeric(as.character(COMPL_RPY_3YR_RT)),
         GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P6 = as.numeric(as.character(MD_EARN_WNE_P6)),
         MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
         MD_EARN_WNE_P10 = as.numeric(as.character(MD_EARN_WNE_P10)),
         CDR3 = as.numeric(as.character(CDR3)), # three-year cohort default rate
         MEDIAN_HH_INC = as.numeric(as.character(MEDIAN_HH_INC)), #Median household income
         AGE_ENTRY = as.numeric(as.character(AGE_ENTRY)),
         UGDS = as.numeric(as.character(UGDS)),
         CONTROL =as.character(CONTROL),
         COSTT4_A = as.numeric(as.character(COSTT4_A)), #avg cost of attendance for academic year insti
         COSTT4_P = as.numeric(as.character(COSTT4_P)), #avg cost of attendance for program year instit
         PCTFLOAN = as.numeric(as.character(PCTFLOAN)),
         PCTPELL = as.numeric(as.character(PCTPELL)),
         MD_FAMINC = as.numeric(as.character(MD_FAMINC)))

college_09_15_valid <- college_09_15_valid_ %>%
  select(INSTNM, COMPL_RPY_3YR_RT, GRAD_DEBT_MDN,PCTFLOAN, PCTPELL, MD_EARN_WNE_P6, MD_EARN_WNE_P8, MD_
         CDR3, MEDIAN_HH_INC, AGE_ENTRY, UGDS, CONTROL, COSTT4_A, COSTT4_P, Year, DEBT_TO_EARN, MD_FAMI
  mutate(COMPL_RPY_3YR_RT = as.numeric(as.character(COMPL_RPY_3YR_RT)),
         GRAD_DEBT_MDN = as.numeric(as.character(GRAD_DEBT_MDN)),
         MD_EARN_WNE_P6 = as.numeric(as.character(MD_EARN_WNE_P6)),
```

```
        MD_EARN_WNE_P8 = as.numeric(as.character(MD_EARN_WNE_P8)),
        MD_EARN_WNE_P10 = as.numeric(as.character(MD_EARN_WNE_P10)),
        CDR3 = as.numeric(as.character(CDR3)), # three-year cohort default rate
        MEDIAN_HH_INC = as.numeric(as.character(MEDIAN_HH_INC)), #Median household income
        AGE_ENTRY = as.numeric(as.character(AGE_ENTRY)),
        UGDS = as.numeric(as.character(UGDS)),
        CONTROL =as.character(CONTROL),
        COSTT4_A = as.numeric(as.character(COSTT4_A)), #avg cost of attendance for academic year insti
        COSTT4_P = as.numeric(as.character(COSTT4_P)), #avg cost of attendance for program year instit
        PCTFLOAN = as.numeric(as.character(PCTFLOAN)),
        PCTPELL = as.numeric(as.character(PCTPELL)),
        MD_FAMINC = as.numeric(as.character(MD_FAMINC)))
```

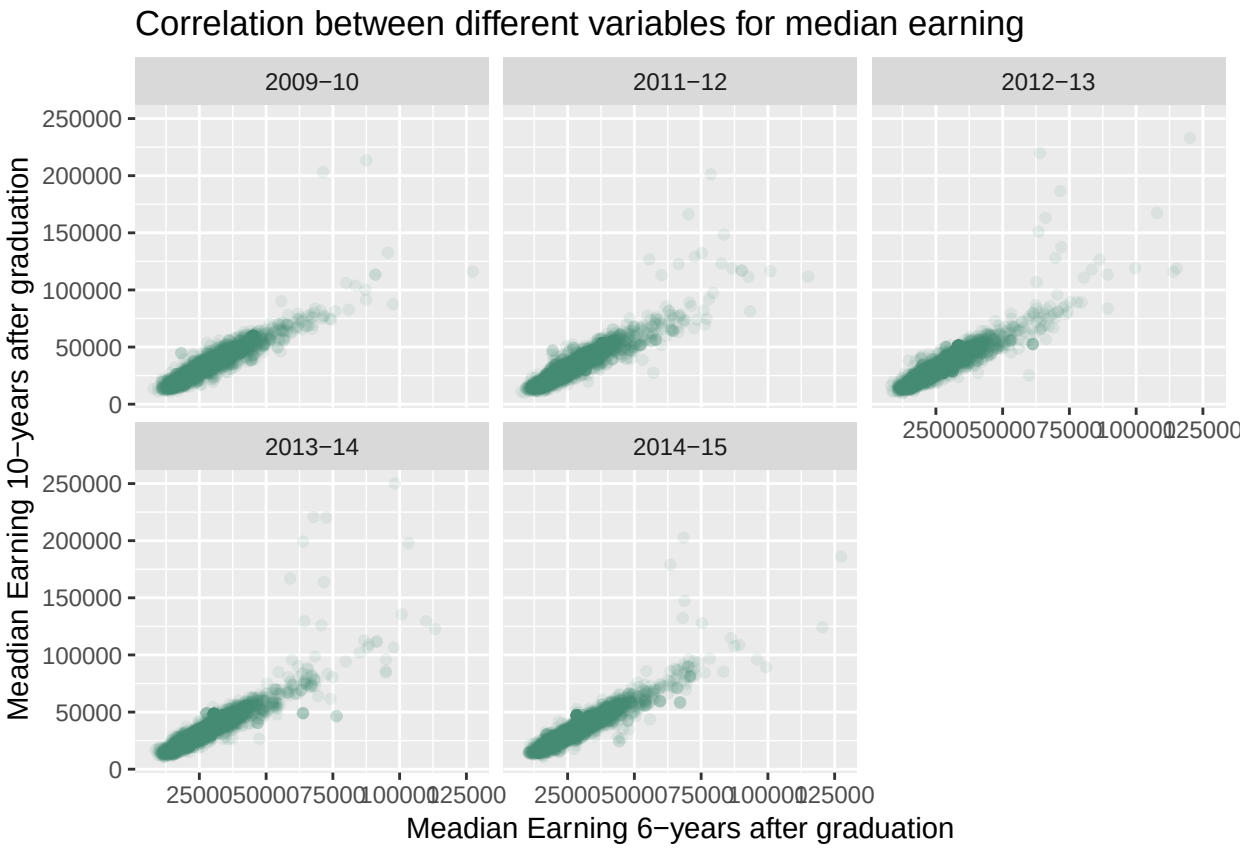
4. EDA for deciding predictors for prediction of response variable: COMPL_RPY_3YR_RT

We have multiple variables related to earnings, such as MD_EARN_WNE_P6, MD_EARN_WNE_P8 and MD_EARN_WNE_P10. We can see if there's a strong correlation between these three, if there is, then we can use only one of them in our modelling.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = MD_EARN_WNE_P6, y = MD_EARN_WNE_P8), color = "darkolivegreen4", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Correlation between different variables for median earning",
       x = "Meadian Earning 6-years after graduation",
       y = "Meadian Earning 8-years after graduation")
```

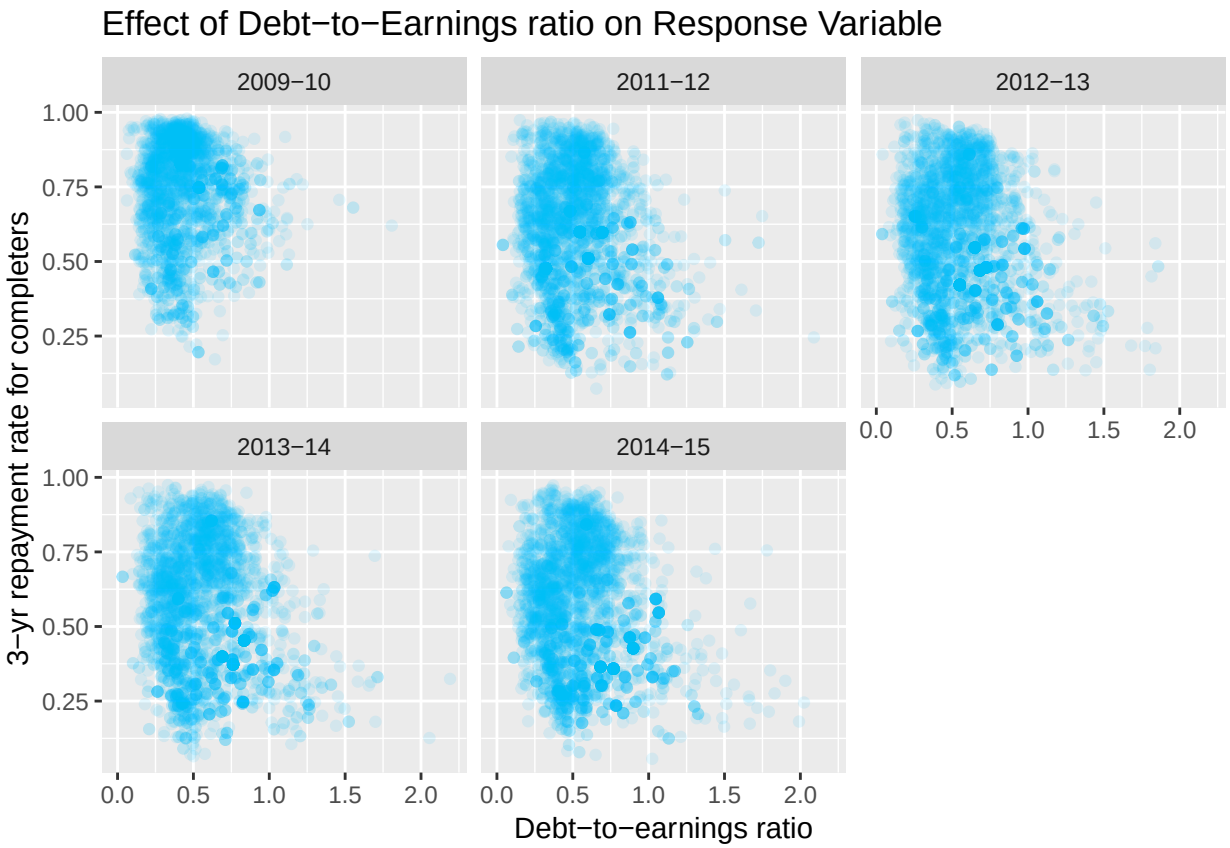



```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = MD_EARN_WNE_P6, y = MD_EARN_WNE_P10), color = "aquamarine4", alpha = 0.1) +
  facet_wrap(~Year) +
  labs(title = "Correlation between different variables for median earning",
       x = "Meadian Earning 6-years after graduation",
       y = "Meadian Earning 10-years after graduation")
```



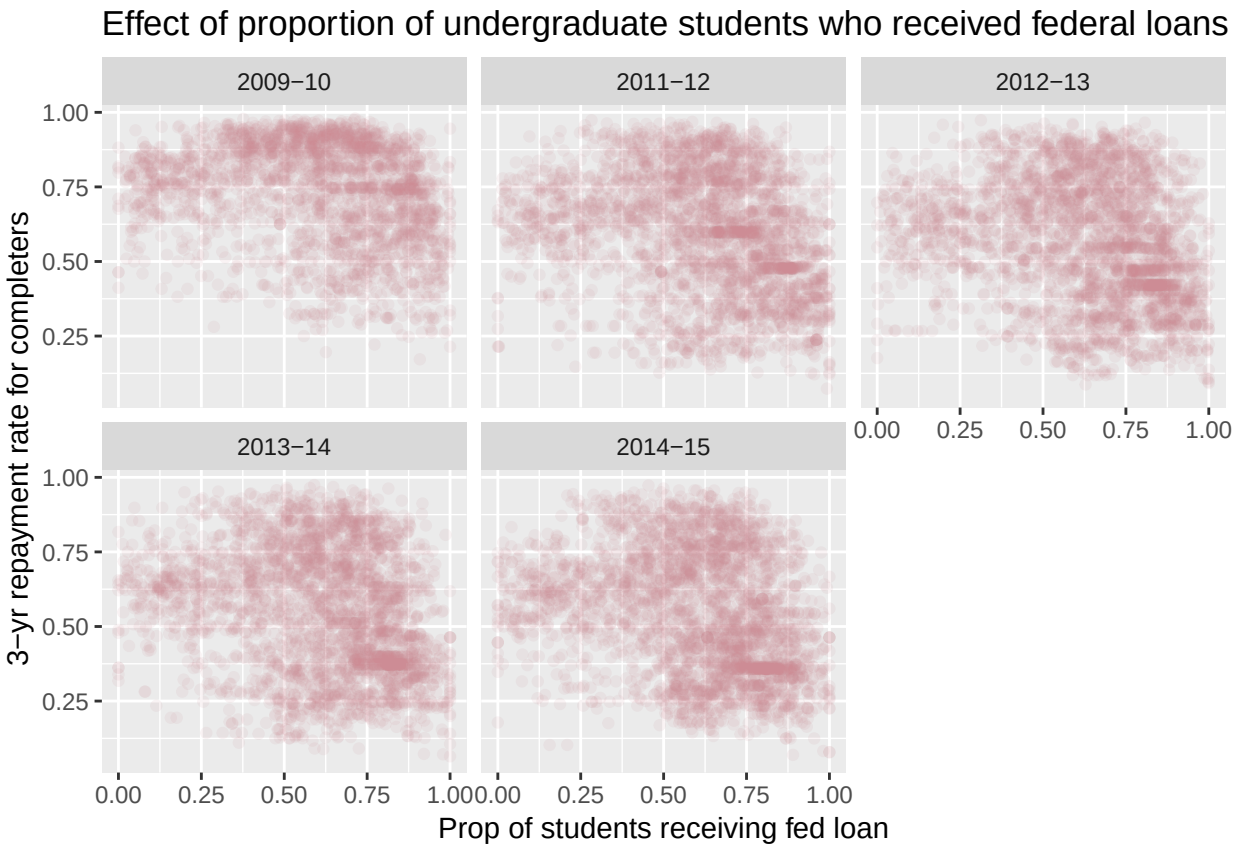
Since we see a high linear correlation between these three, we can arbitrarily decide to keep MD_EARN_WNE_P8 for our predictor model.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = DEBT_TO_EARN, y = COMPL_RPY_3YR_RT), color = "deepskyblue", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Effect of Debt-to-Earnings ratio on Response Variable",
       x = "Debt-to-earnings ratio",
       y = "3-yr repayment rate for completers")
```



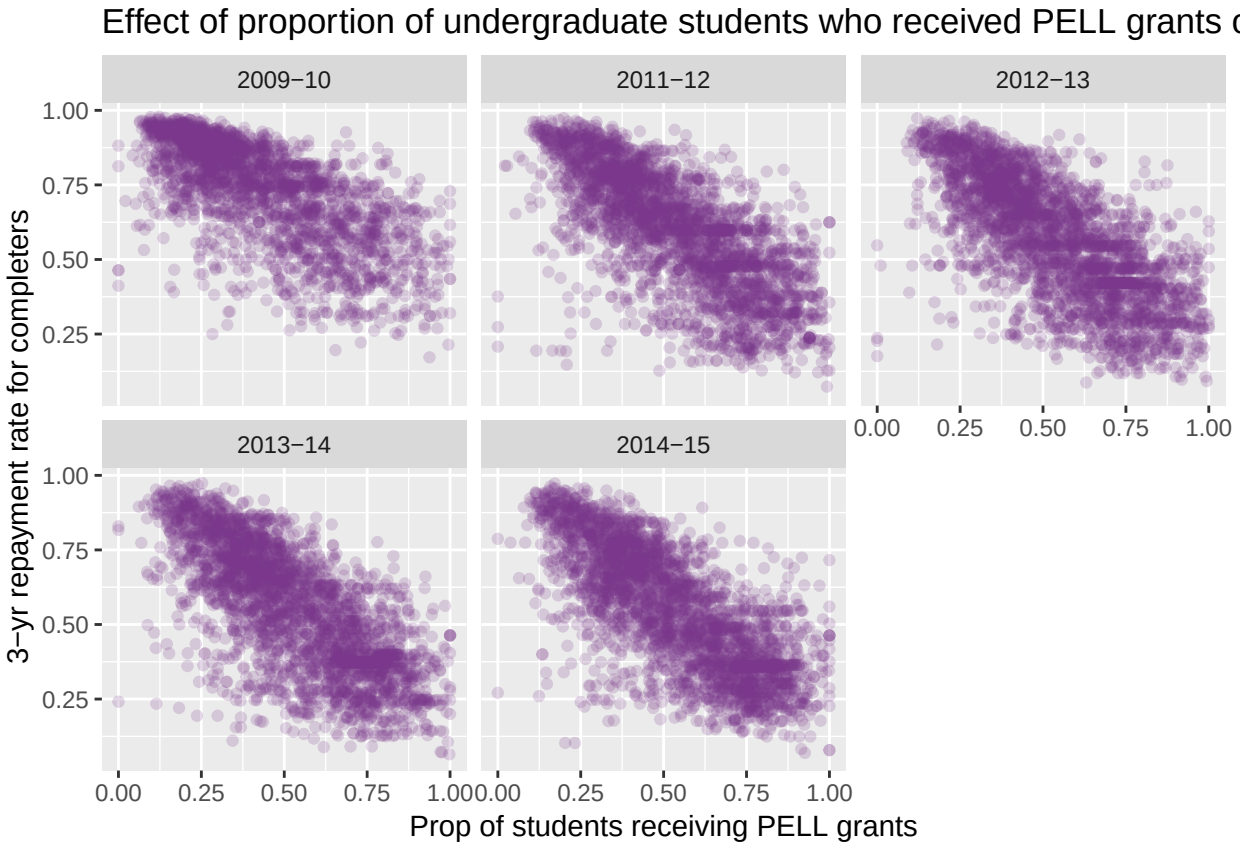
We observe some sort of negative correlation, we can keep DEBT_TO_EARN ration for our prediction.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = PCTFLOAN, y = COMPL_RPY_3YR_RT), color = "lightpink3", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Effect of proportion of undergraduate students who received federal loans on Response Va
       x = "Prop of students receiving fed loan",
       y = "3-yr repayment rate for completers")
```



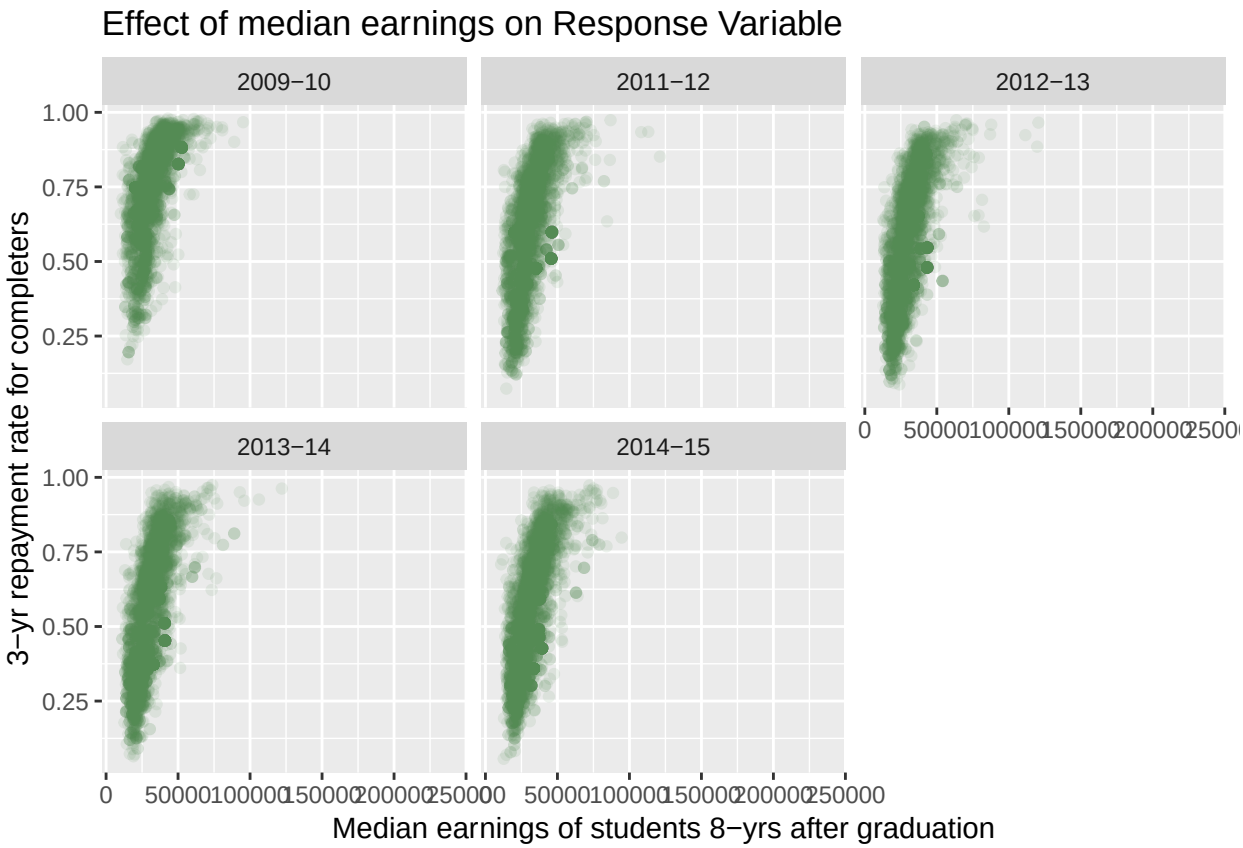
Mostly random, disregard.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = PCTPELL, y = COMPL_RPY_3YR_RT), color = "mediumorchid4", alpha = 0.2) +
  facet_wrap(~Year)+
  labs(title = "Effect of proportion of undergraduate students who received PELL grants on Response Vari
       x = "Prop of students receiving PELL grants",
       y = "3-yr repayment rate for completers")
```



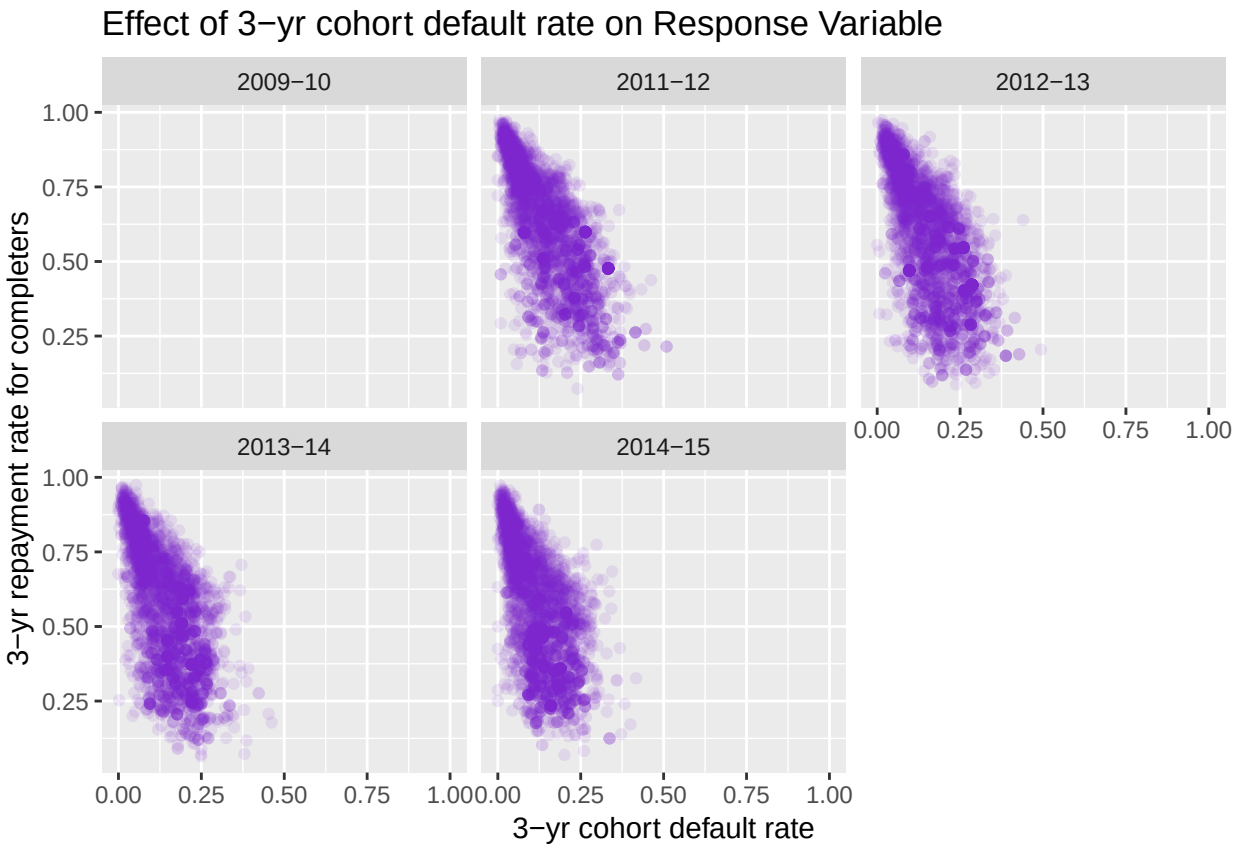
Strong negative correlation, keep.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = MD_EARN_WNE_P8, y = COMPL_RPY_3YR_RT), color = "palegreen4", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Effect of median earnings on Response Variable",
       x = "Median earnings of students 8-yrs after graduation",
       y = "3-yr repayment rate for completers")
```



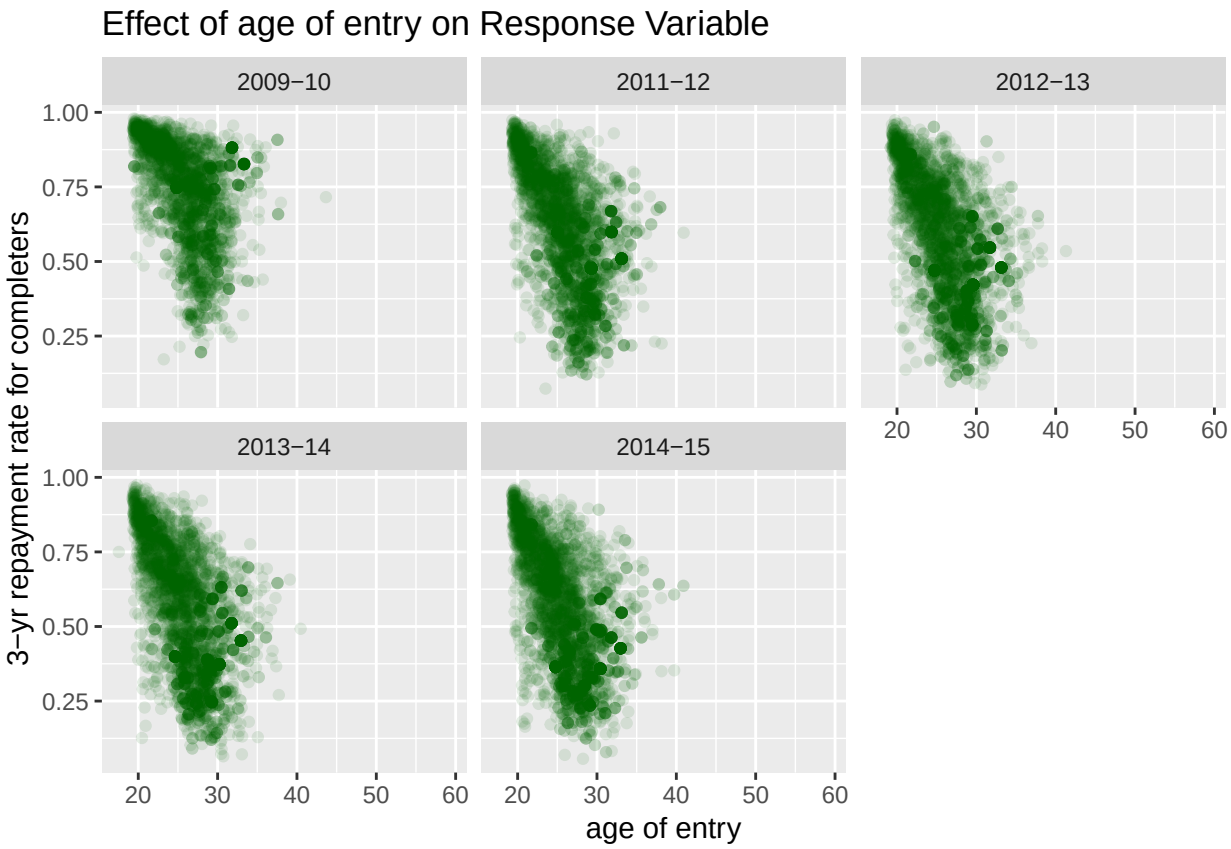
Keep MD_EARN_WNE_P8.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = CDR3, y = COMPL_RPY_3YR_RT), color = "purple3", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Effect of 3-yr cohort default rate on Response Variable",
       x = "3-yr cohort default rate",
       y = "3-yr repayment rate for completers")
```



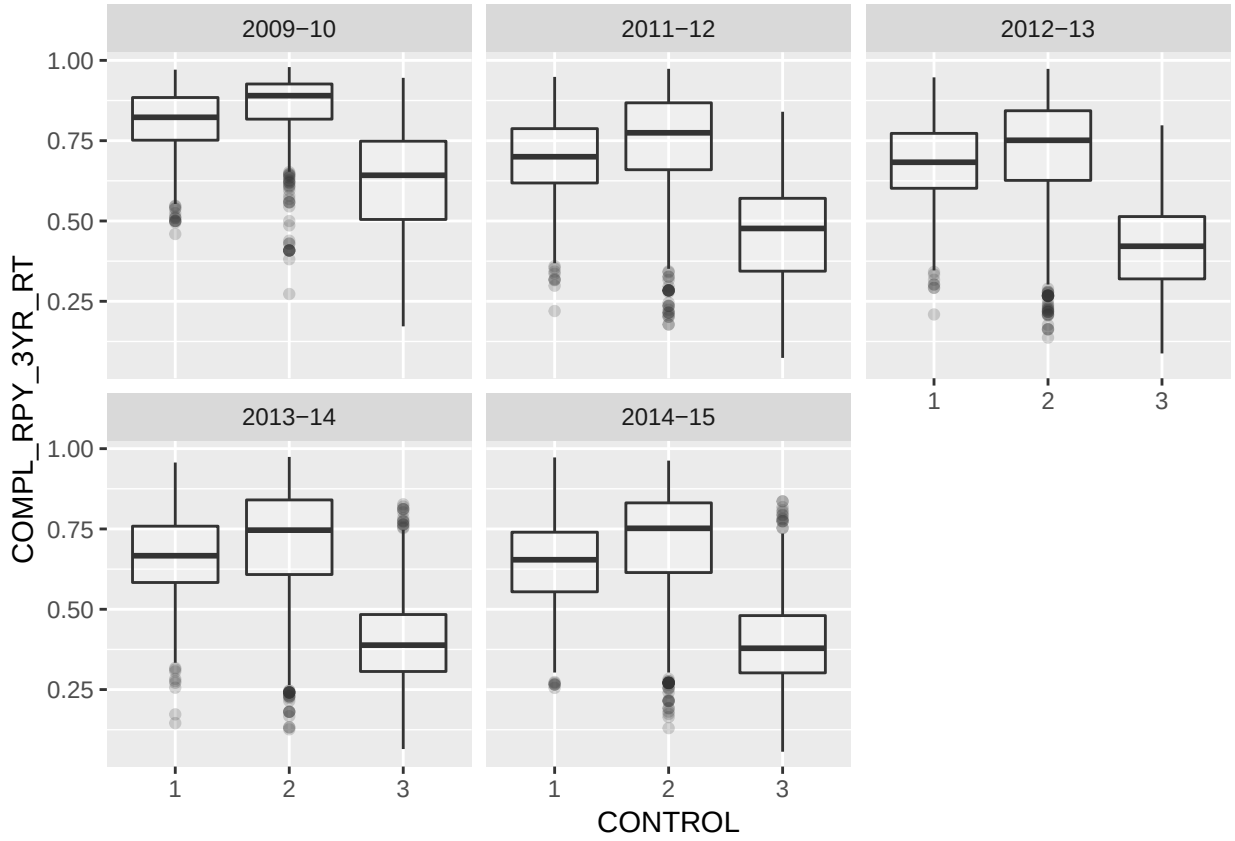
Keep CDR3.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = AGE_ENTRY, y = COMPL_RPY_3YR_RT), color = "darkgreen", alpha = 0.1) +
  facet_wrap(~Year)+
  labs(title = "Effect of age of entry on Response Variable",
       x = "age of entry",
       y = "3-yr repayment rate for completers")
```



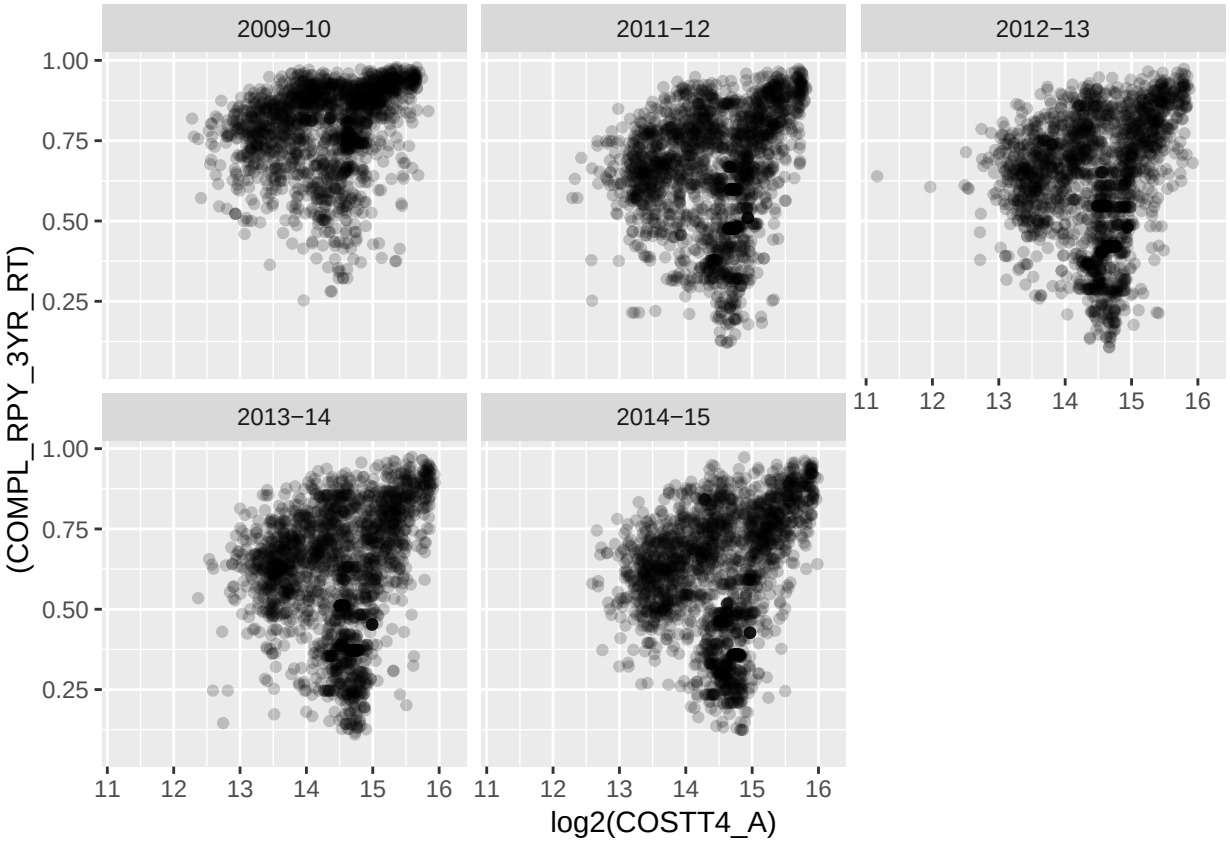
Keep AGE_ENTRY.

```
college_09_15_train %>%
  ggplot() +
  geom_boxplot(aes(x = CONTROL, y = COMPL_RPY_3YR_RT), alpha = 0.2) +
  facet_wrap(~Year)
```

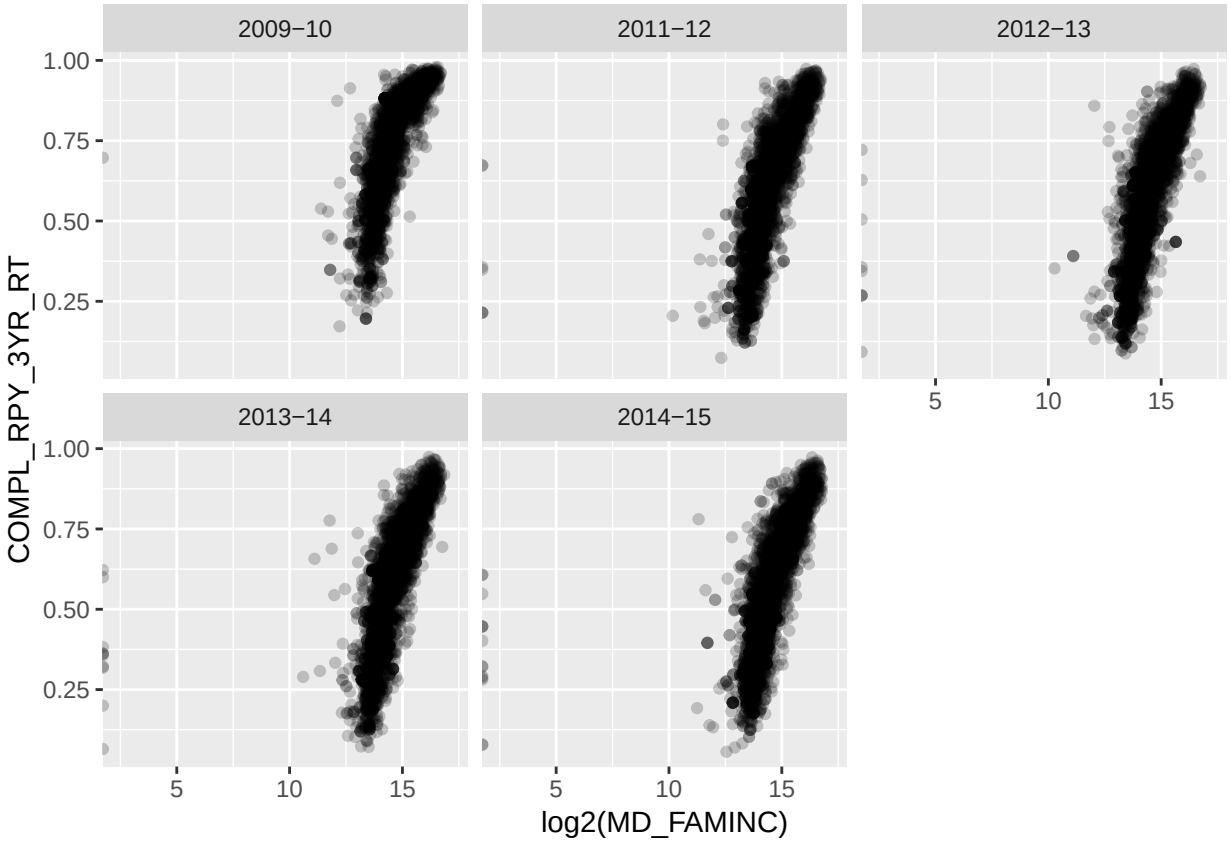


wide fluctuation among control groups, so keeping CONTROL as one of the predictors.

```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = log2(COSTT4_A), y = (COMPL_RPY_3YR_RT)), alpha = 0.2) +
  facet_wrap(~Year)
```




```
college_09_15_train %>%
  ggplot() +
  geom_point(aes(x = log2(MD_FAMINC), y = COMPL_RPY_3YR_RT), alpha = 0.2) +
  facet_wrap(~Year)
```



Keeping MD_FAMINC.

5. Now we can try fitting a linear model with the above predictor variables.

```
set.seed(1)

college_09_15_train <- college_09_15_train %>%
  filter(Year != "2009-10") %>%
  mutate(log_MD_FAMINC = log2(MD_FAMINC)) %>%
  filter(log_MD_FAMINC >= 0)

college_09_15_test <- college_09_15_test %>%
  filter(Year != "2009-10") %>%
  mutate(log_MD_FAMINC = log2(MD_FAMINC)) %>%
  filter(log_MD_FAMINC >= 0)

college_09_15_valid <- college_09_15_valid %>%
  filter(Year != "2009-10") %>%
  mutate(log_MD_FAMINC = log2(MD_FAMINC)) %>%
  filter(log_MD_FAMINC >= 0)


# this model was decided after trying various combinations of predictors, checking their R-squared valu
# the analysis of this can be found later in this document in the 6th section titled: "R-Square analysi
model <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN + PCTPELL  + MD_EARN_WNE_P8 + log2(COSTT4_A) + log_MD_FAMINC
```

```
              data = college_09_15_train)

summary(model)
```

```
##
## Call:
## lm(formula = COMPL_RPY_3YR_RT ~ DEBT_TO_EARN + PCTPELL + MD_EARN_WNE_P8 +
##     log2(COSTT4_A) + log_MD_FAMINC, data = college_09_15_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43056 -0.05737  0.00109  0.05823  0.52098
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -8.596e-01  3.398e-02  -25.30   <2e-16 ***
## DEBT_TO_EARN   -7.432e-02  5.072e-03  -14.65   <2e-16 ***
## PCTPELL        -1.848e-01  8.135e-03  -22.72   <2e-16 ***
## MD_EARN_WNE_P8  2.537e-06  1.604e-07   15.82   <2e-16 ***
## log2(COSTT4_A) -3.462e-02  2.081e-03  -16.64   <2e-16 ***
## log_MD_FAMINC   1.382e-01  2.090e-03   66.13   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09216 on 7970 degrees of freedom
##   (9650 observations deleted due to missingness)
## Multiple R-squared:  0.7673, Adjusted R-squared:  0.7672
## F-statistic:  5256 on 5 and 7970 DF,  p-value: < 2.2e-16
```

```
print("RMSE for train data")
```

```
## [1] "RMSE for train data"
```

```
rmse(model, college_09_15_train) #0.092267
```

```
## [1] 0.09212411
```

```
print("RMSE for test data")
```

```
## [1] "RMSE for test data"
```

```
rmse(model, college_09_15_test)  #0.09390385
```

```
## [1] 0.09371103
```

```
print("Mean Absolute Error for Test Data" )
```

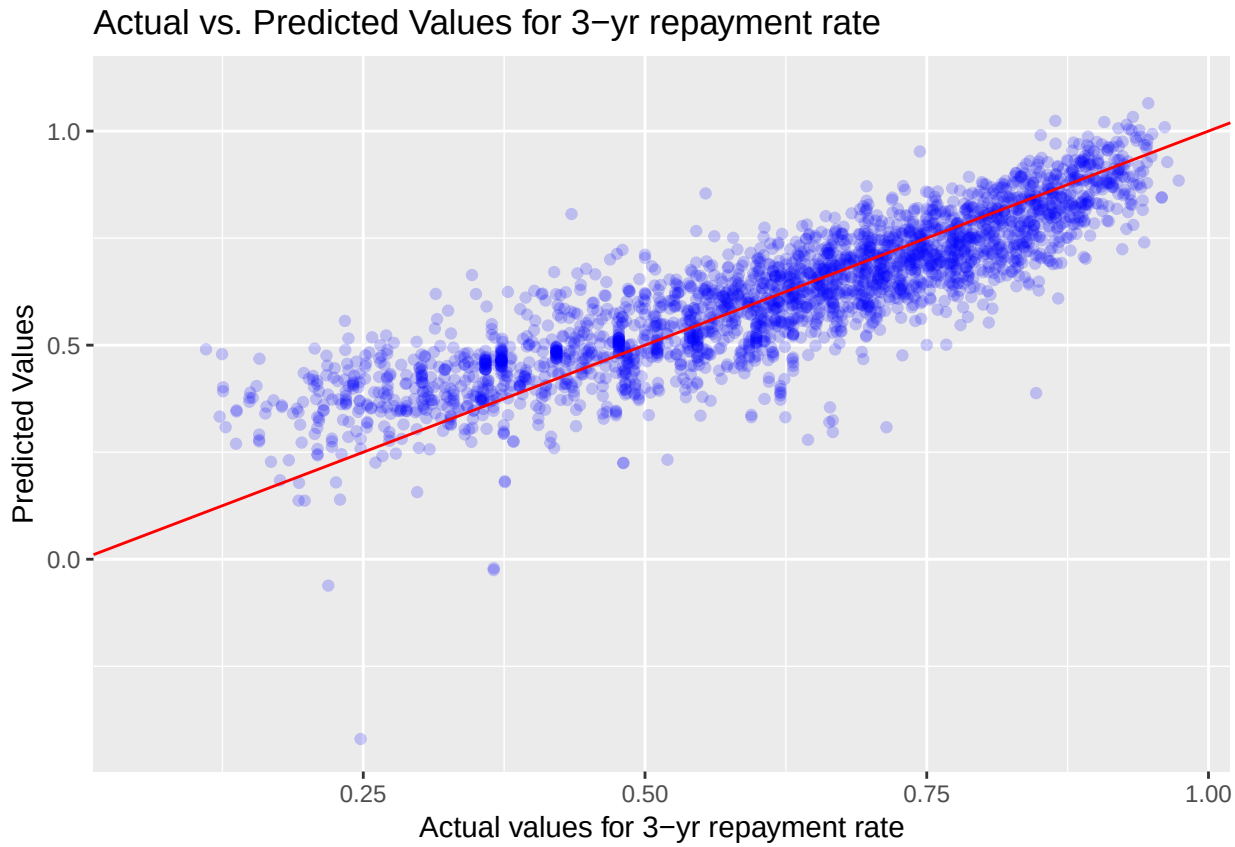
```
## [1] "Mean Absolute Error for Test Data"
```

```
mae(model, college_09_15_test)
```

```
## [1] 0.07171087
```

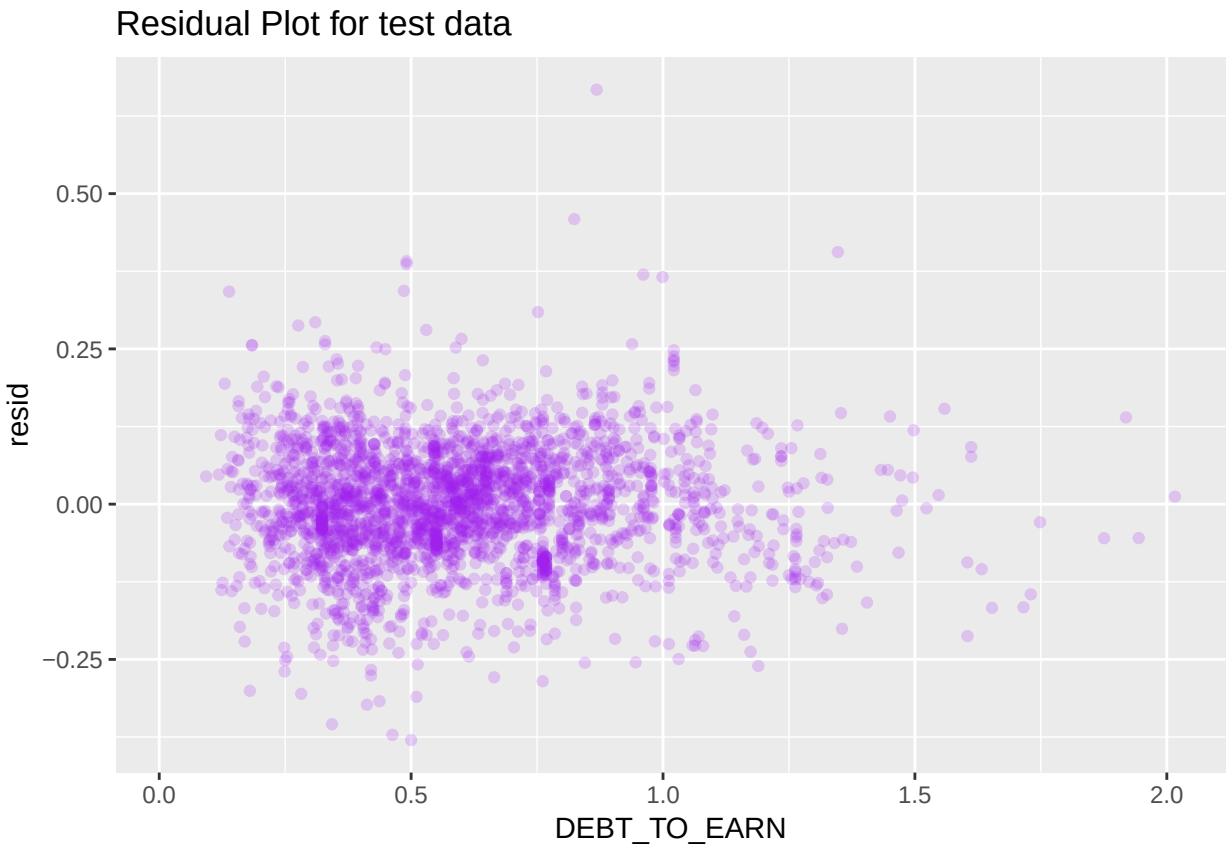
Plotting the actual vs predicted data for test set:

```
college_09_15_test %>%
  add_predictions(model) %>%
  ggplot(aes(x=COMPL_RPY_3YR_RT, y = pred)) +
  geom_point(color = "blue", alpha = 0.2)+
  geom_abline(color = "red") +
  labs(title = "Actual vs. Predicted Values for 3-yr repayment rate",
       x = "Actual values for 3-yr repayment rate",
       y = "Predicted Values")
```

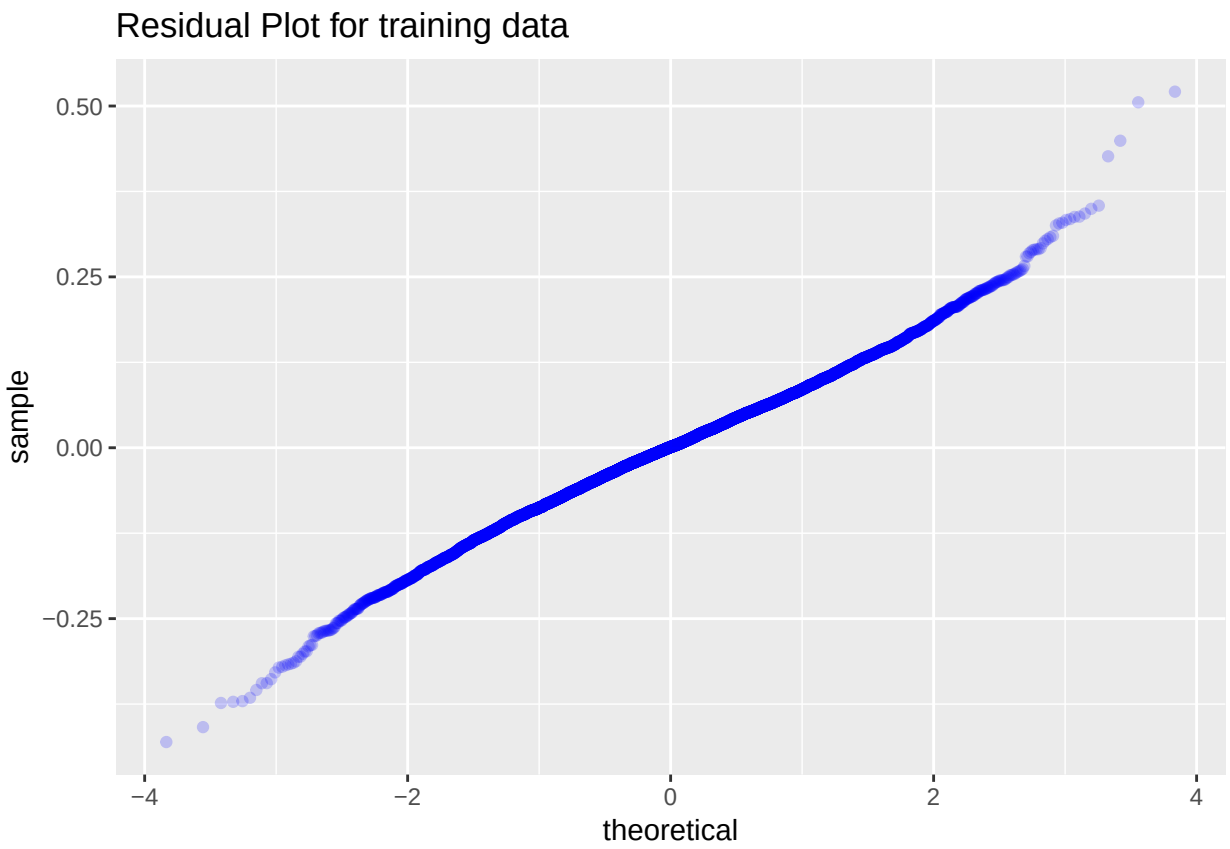


Plotting the residuals for the training data:

```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=DEBT_TO_EARN, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```

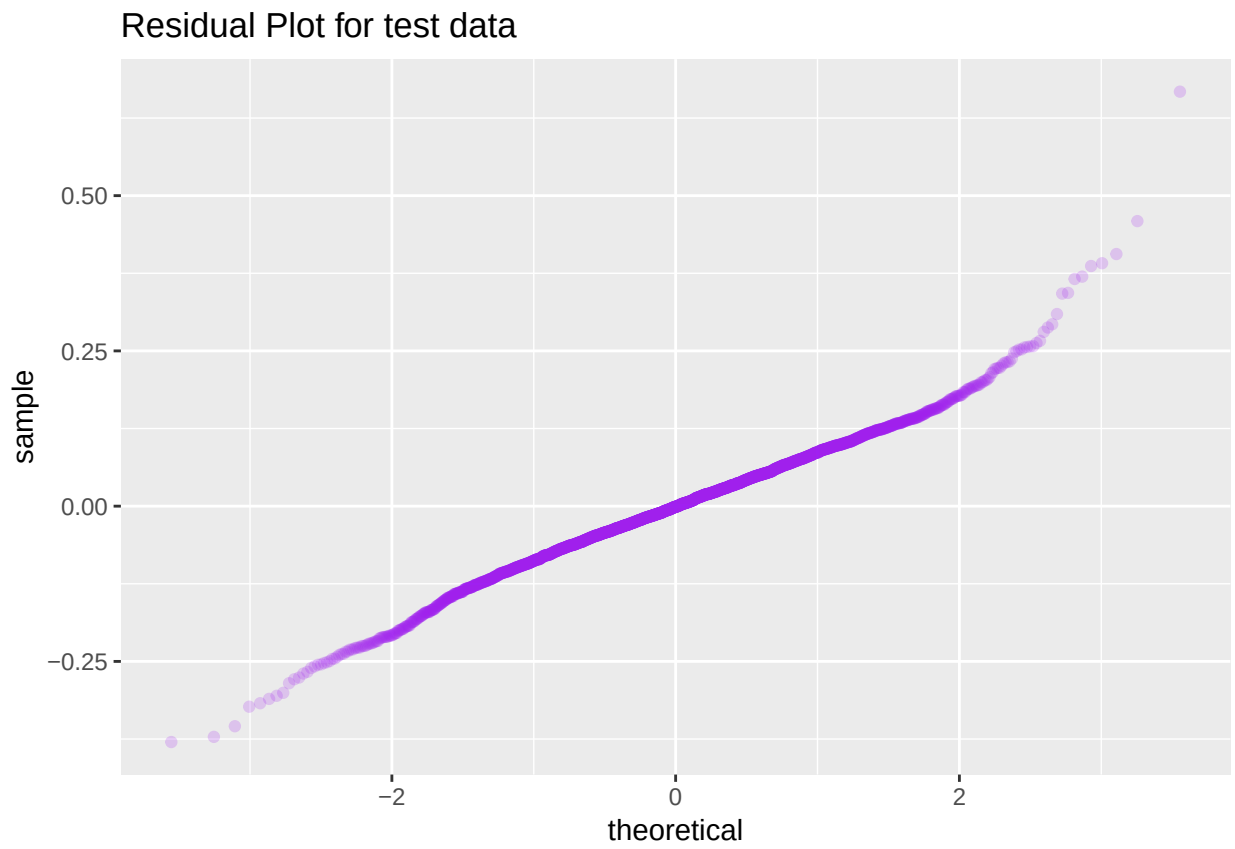



```
college_09_15_train %>%
  add_residuals(model) %>%
  ggplot(aes(sample=resid)) +
  geom_qq(color = "blue", alpha = 0.2) +
  labs(title = "Residual Plot for training data")
```

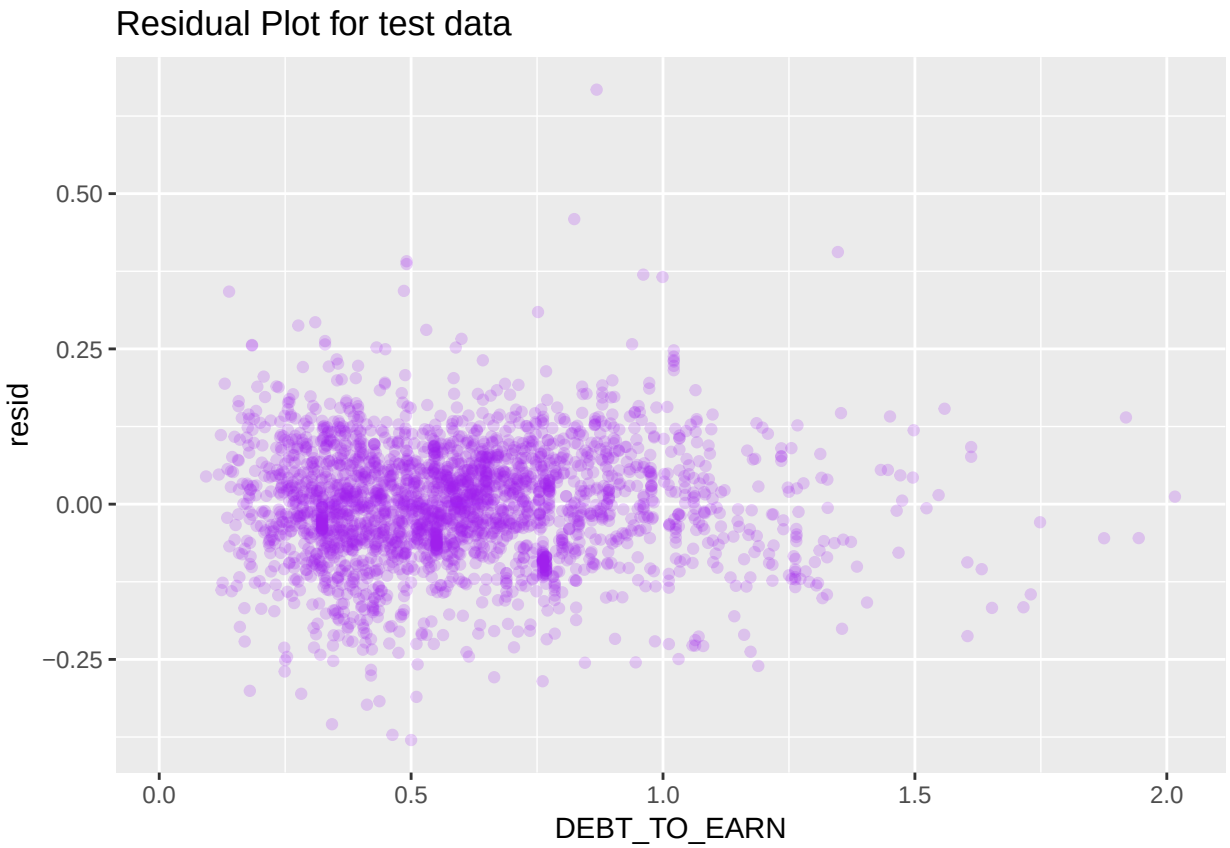


Plotting the residuals for the test data:

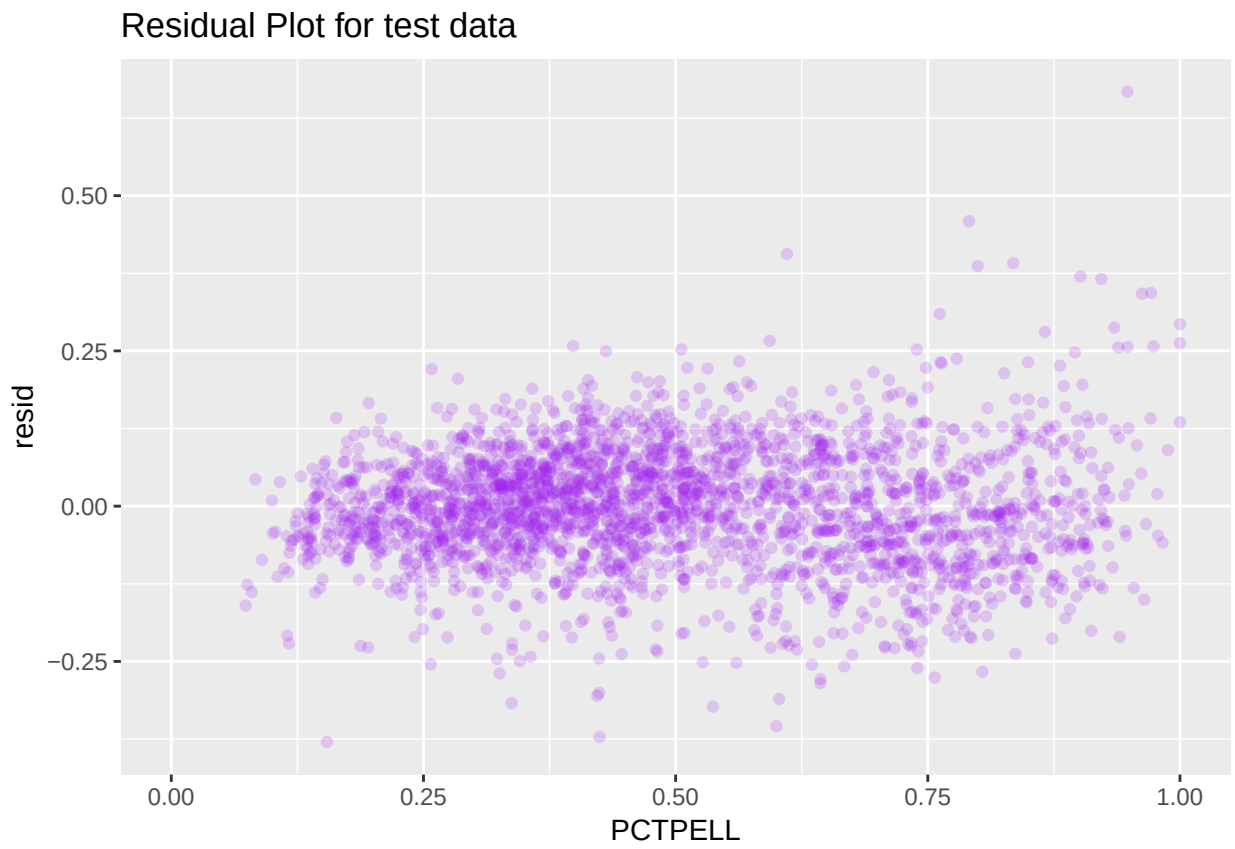
```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(sample=resid)) +
  geom_qq(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```



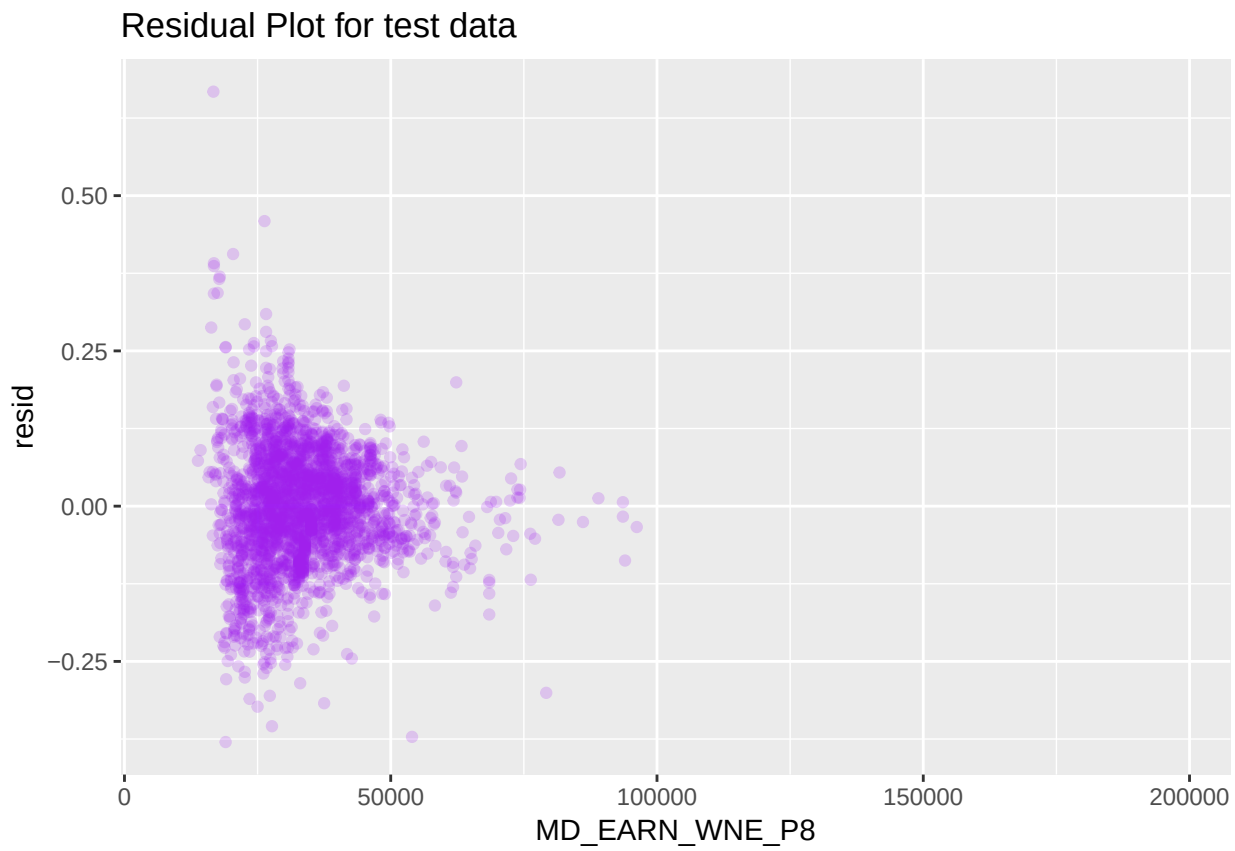

```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=DEBT_TO_EARN, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```



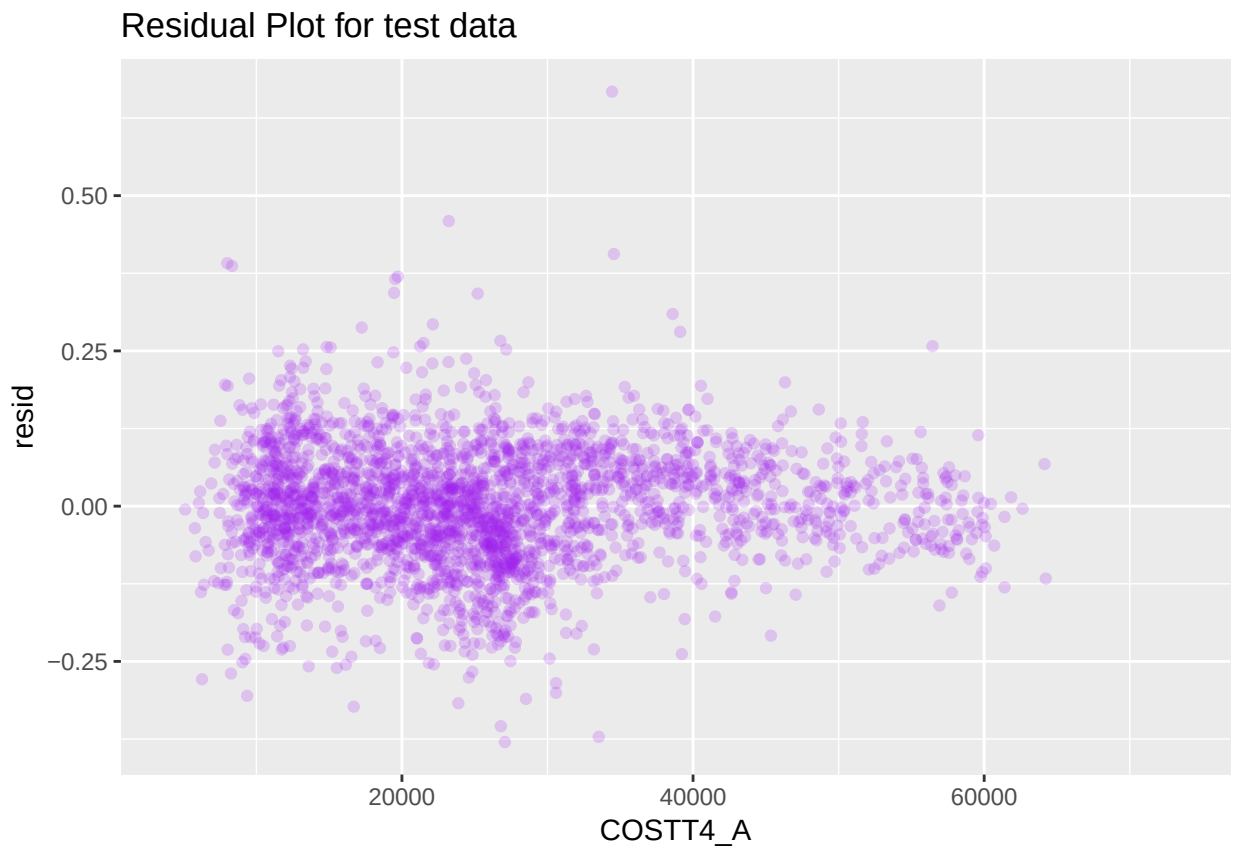

```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=PCTPELL, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```



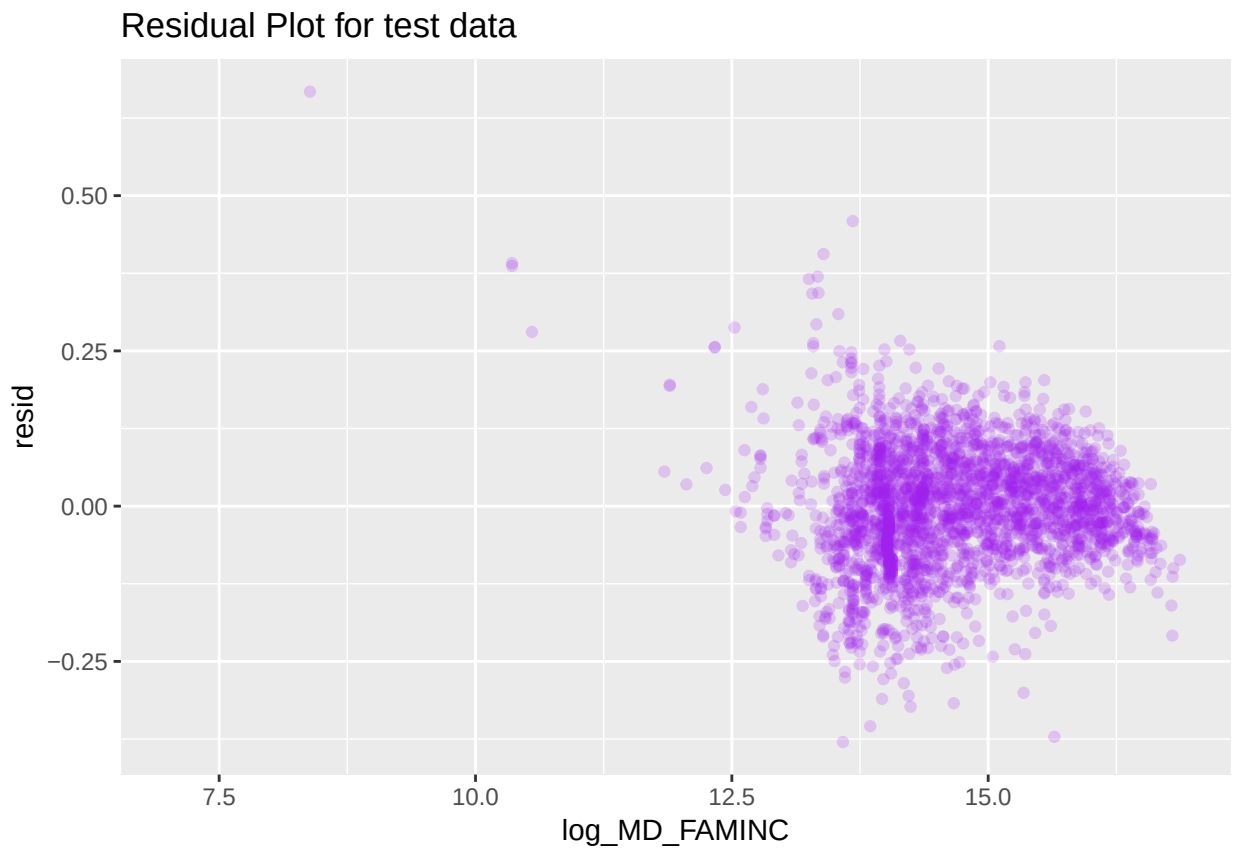

```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=MD_EARN_WNE_P8, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```




```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=COSTT4_A, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```




```
college_09_15_test %>%
  add_residuals(model) %>%
  ggplot(aes(x=log_MD_FAMINC, y = resid)) +
  geom_point(color = "purple", alpha = 0.2)+
  labs(title = "Residual Plot for test data")
```



Set of 10 observations for the response variable (3-yr repayment rate for completers), vs the predicted values returned by the model.

```
college_09_15_test %>%
  add_predictions(model) %>%
  filter(!is.na(pred),!is.na(COMPL_RPY_3YR_RT) ) %>%
  select(INSTNM, COMPL_RPY_3YR_RT, pred) %>%
  transmute("Institute Name" = INSTNM,
            "3-yr Repayment Rate - Actual values" = COMPL_RPY_3YR_RT,
            "Predicted Values" = pred) %>%
  head(10)
```

```
## # A tibble: 10 x 3
##    `Institute Name`              `3-yr Repayment Rate - ~ `Predicted Valu~
##    <fct>                                            <dbl>            <dbl>
##  1 Auburn University                                0.879            0.905
##  2 George C Wallace State Commu~                    0.626            0.627
##  3 Huntingdon College                               0.75             0.713
##  4 Marion Military Institute                        0.606            0.775
##  5 University of Mobile                             0.610            0.633
##  6 Southeastern Bible College                       0.621            0.553
##  7 Stillman College                                 0.3              0.446
##  8 University of Alaska Southea~                    0.747            0.746
##  9 Arkansas State University-Ma~                    0.722            0.610
## 10 Arkansas Tech University                         0.710            0.649
```

6.R-Square analysis for overfitting tests.

```
 r_square_values <- function(linear.model) {
  r_squared <- summary(linear.model)$r.squared
  adjusted_r_squared <- summary(linear.model)$adj.r.squared
  predicted_r_squared <- predicted_r_squared_val(linear.model)
  return.df <- data.frame(r.squared = r_squared, adjusted.r.squared = adjusted_r_squared, predicted.r.s
  return(return.df)
}

predicted_r_squared_val <- function(linear.model) {
  lm.anova <- anova(linear.model)
  tot_sum_sq <- sum(lm.anova$'Sum Sq')
  pred.r.squared <- 1-pred_residual_sq_sum(linear.model)/(tot_sum_sq)

  return(pred.r.squared)
}

pred_residual_sq_sum <- function(linear.model) {
  pred <- residuals(linear.model)/(1-lm.influence(linear.model)$hat)
  pred_resid_sum <- sum(pred^2)

  return(pred_resid_sum)
}
```

```
model_1 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN,
              data = college_09_15_train)
rmse(model_1, college_09_15_valid) #0.1986109
```

```
## [1] 0.1982403
```

```
model_2 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN, + MD_EARN_WNE_P8,
              data = college_09_15_train)
rmse(model_2, college_09_15_valid) #0.199981
```

```
## [1] 0.2135804
```

```
model_3 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN, + MD_EARN_WNE_P8 + PCTPELL,
              data = college_09_15_train)
rmse(model_3, college_09_15_valid) #0.1997196
```

```
## [1] 0.214152
```

```
model_4 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN + PCTPELL  + MD_EARN_WNE_P8 + log2(COSTT4_A) + log_MD_FAM
              data = college_09_15_train)
rmse(model_4, college_09_15_valid) #0.0921737
```

```
## [1] 0.09282367
```

```
model_5 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN, + MD_EARN_WNE_P8 +PCTPELL + CDR3 + log2(COSTT4_A),
              data = college_09_15_train) #0.2250866
rmse(model_5, college_09_15_valid)
```

```
## [1] 0.2071611
```

```
model_6 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN, + MD_EARN_WNE_P8 +PCTPELL + CDR3 + log2(COSTT4_A) + log_
              data = college_09_15_train)
rmse(model_6, college_09_15_valid) #0.2343732
```

```
## [1] 0.2562501
```

```
model_7 <- lm(COMPL_RPY_3YR_RT ~ DEBT_TO_EARN, + MD_EARN_WNE_P8 +PCTPELL + CDR3 + log2(COSTT4_A) + log_
              data = college_09_15_train)
rmse(model_7, college_09_15_valid) #0.2104708
```

```
## [1] 0.205953
```

```
ldply(list(model_1, model_2, model_3, model_4, model_5, model_6, model_7), r_square_values)
```

```
##     r.squared adjusted.r.squared predicted.r.squared
## 1 0.032192616        0.032111722          0.03190828
## 2 0.024525039        0.023661786          0.02135378
## 3 0.021968370        0.021018824          0.01844675
## 4 0.767311449        0.767165472          0.76688293
## 5 0.088343611        0.078945092          0.05281721
## 6 0.006301397       -0.004990633         -0.03299861
## 7 0.136031326        0.128646979          0.10933740
```

```
rmse(model_4, college_09_15_test)
```

```
## [1] 0.09371103
```