Technical Appendix A: Initial EDA

Data Wrangling

```r
# Library packages
library(plyr)
library(dplyr)
library(ggplot2)
library(readxl)
library(rvest)
library(robotstxt)
library(tidyverse)
```

Data Sets for Figure 1-4 & Figure 6

```r
# Import scholarship data
scholarship <- read_excel("DR-826 Pittsburgh Promise scholarship use.xlsx", sheet = "Sheet1")

# Clean scholarship data
names(scholarship)[1]<-"RandomID"
scholarship$GradYear <- as.character(scholarship$GradYear)
```

```r
# Import demographics data
demo1415 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1415")
demo1516 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1516")
demo1617 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1617")

# Clean demographics data
demo1416 <- join(demo1415,demo1516,type="full")
demo <- join(demo1416,demo1617,type="full")

# Filter observations for students' senior year
demo_senior <- demo %>%
  select(-c("Cohort")) %>%
  mutate(GradYear = substr(as.character(SchoolYear),5,8)) %>%
  filter(GradeCode == 12)
demo_senior$RandomID <- as.character(demo_senior$RandomID)
```

```r
# Join demographic data in senior year and scholarship data together
# Only senior year since qualification is evaluated after graduating from high school
demo_scholarship <- inner_join(demo_senior, scholarship, by = c("RandomID", "GradYear"))

# Filter repetitive observations out
demo_scholarship_final <- distinct(demo_scholarship) #2223 obs in total
```

Data Sets for Figure 5

```r
# Import CTE data
CTE <-
  read_excel("DR-837 Career and Tech. Education (CTE) outcomes (industrial-recognized certificates).xlsx",
             sheet = "CTE data")

# Generate column that calculates the total number of certifications earned by each student
cte_id_check <- CTE %>%
  group_by(RandomID) %>%
  dplyr::summarise(num_cte = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
# Generate new CTE data that includes a column showing the total number of
# certifications earned by each student.
CTE_new = left_join(CTE, cte_id_check, by = "RandomID")
```

```r
# Manipulate CTE and Scholarship data to eliminate to prepare for later joint data set
CTE_new <- CTE_new %>%
  select(-c("Cohort"))
CTE_new$RandomID <- as.character(CTE_new$RandomID)

# Join CTE and Scholarship data
CTE_scholarship = inner_join(scholarship, CTE_new, by = "RandomID")
CTE_scholarship <- distinct(CTE_scholarship)
```

Data Sets for Figure 7 & Figure 8

```r
# Import attendance data
attendance <- read_excel("DR-830 Attendance.xlsx",
    sheet = "Attendance data")
```

```r
# Clean attendance data
attendance_new <- attendance %>%
  mutate_at(c(1, 3:9), as.factor)
names(attendance_new)[10] <- "CountofDays"
# Create attendance_scholarship table: attendance left_join scholarship
scholarship_new <- scholarship %>%
  mutate_at(c(2:8), as.factor)
attendance_scholarship <- attendance_new %>%
  left_join(scholarship_new, by = c("RandomID"="RandomID"))
# Create a long table where sum each student's recorded days based on attendance status
attendance_scholarship_long <- attendance_scholarship %>%
  dplyr::select(RandomID, AttendanceStatus, CountofDays) %>%
  group_by(RandomID, AttendanceStatus) %>%
  dplyr::summarise(TotalDays = sum(CountofDays))
# Create a wide table that each student takes 1 row
attendance_scholarship_wide <- attendance_scholarship_long %>%
  spread(AttendanceStatus, TotalDays) %>%
```

```
  dplyr::select(-'NULL') %>%
  replace(is.na(.), 0)
# Add total count of days
attendance_scholarship_wide$totalDays <- rowSums(attendance_scholarship_wide[2:6])
```

```
# Create attendance_rate data
# attendance data is the joined data set between scholarship and attendance
wide_scholarship <- attendance_scholarship_wide %>%
  left_join(scholarship_new, by = c("RandomID"="RandomID")) %>%
  na.omit(.)
# Add excused percentage
wide_scholarship <- wide_scholarship %>%
  mutate(excused_pct = (`Present`+`Present Excused`+`Present Unexcused`+`Absent Excused`)/totalDays)
# Categorize excused absent rate
wide_scholarship <- wide_scholarship %>%
  mutate(excused_pct_cate = if_else(excused_pct >= 0.9, ">=90%", "<90%"))

attendance_rate <- wide_scholarship
```

```
# Import GPA data
gpa <- read_excel("DR-831 GPA.xlsx", sheet = "GPA")
```

```
# Clean GPA data
gpa_no_cohort <- gpa %>% dplyr::select(-Cohort)
gpa_no_cohort <- gpa_no_cohort[!duplicated(gpa_no_cohort),]

# Create senior_gpa data that includes cumulative GPA of each student
senior_students <- (gpa_no_cohort %>%
                     group_by(RandomID) %>%
                     dplyr::summarise(n = n()) %>%
                     filter(n==4))$RandomID
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
senior_gpa <- gpa_no_cohort %>%
  filter(RandomID %in% senior_students) %>%
  group_by(RandomID) %>%
  filter(SchoolYear == max(SchoolYear))
```

Visualizations

Figure 1: Qualification for Promise Scholarship vs. Race

```
ggplot(demo_scholarship_final,
       aes(fill=Race, x=QualifiedforCorePromise)) +
  geom_bar(position="fill") +
  ggtitle("Qualified for Promise and Race") +
  labs(y="Proportion")
```

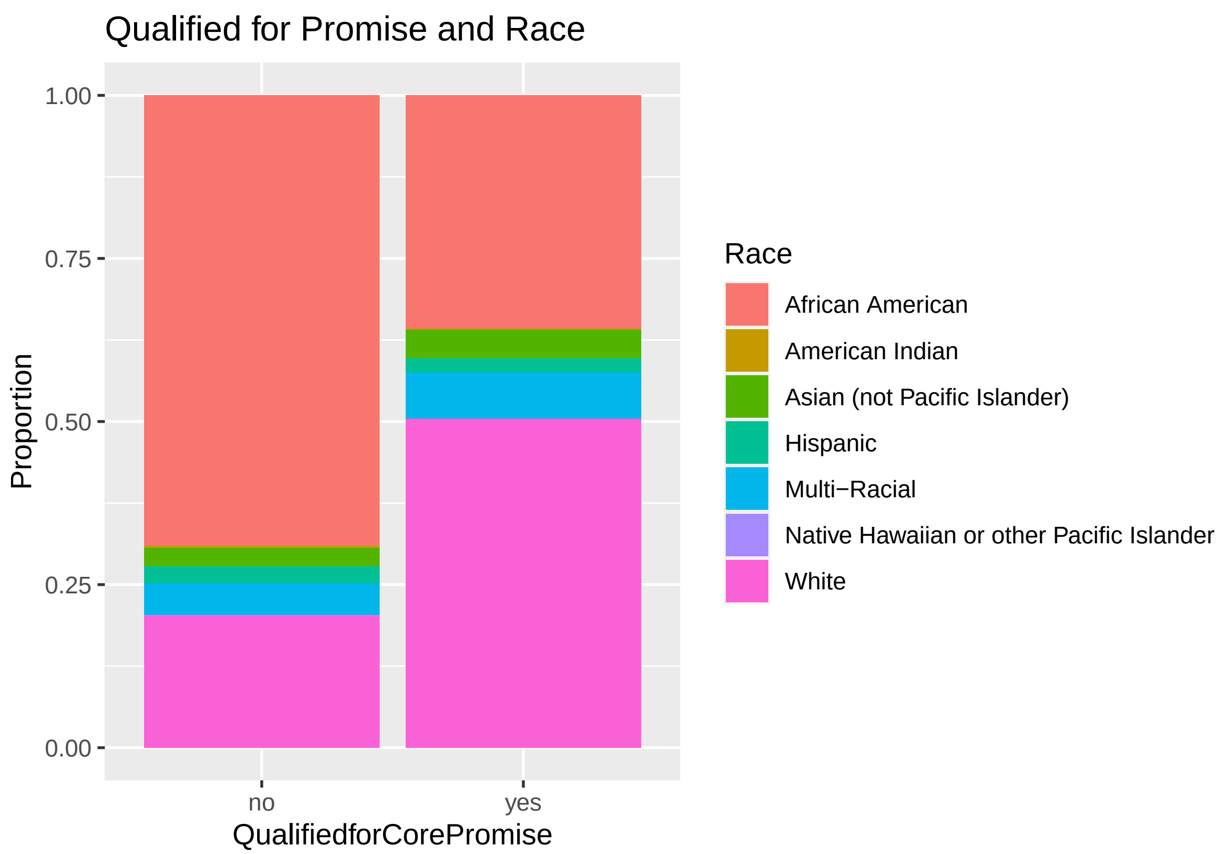


Figure 2: Receipt of Promise Scholarship vs. Race

```
# Filter qualified students in the data set
qualified <- demo_scholarship_final %>%
  filter(QualifiedforCorePromise == "yes") #1560 observations

ggplot(qualified, aes(fill=Race, x=EverReceivedPromiseAward)) +
  geom_bar(position="fill") +
  ggtitle("Ever Received Promise and Race") +
  labs(y="Proportion")
```

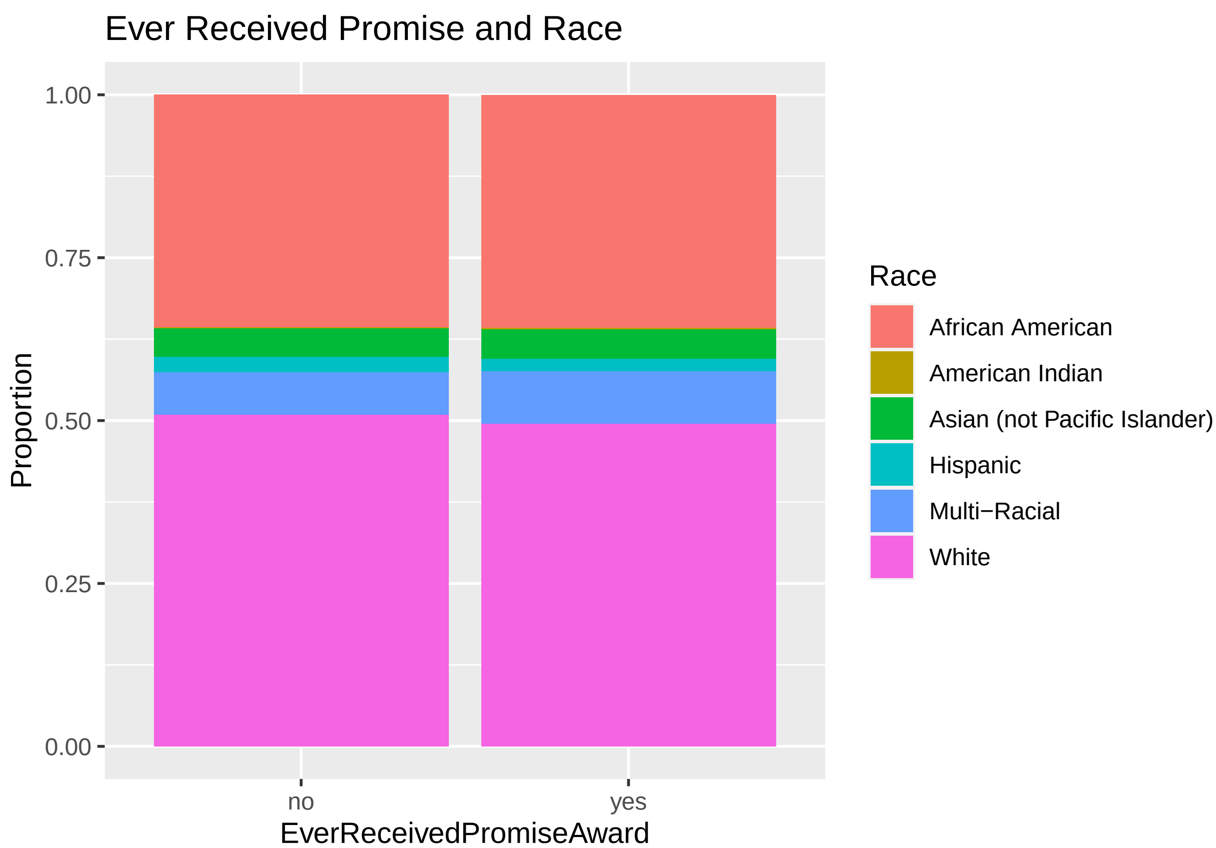


Figure 3: Qualification for Promise Scholarship vs. Gender

```
ggplot(demo_scholarship_final,
       aes(fill=Gender, x=QualifiedforCorePromise)) +
  geom_bar(position="fill") +
  ggtitle("Qualified for Promise and Gender") +
  labs(y="Proportion")
```

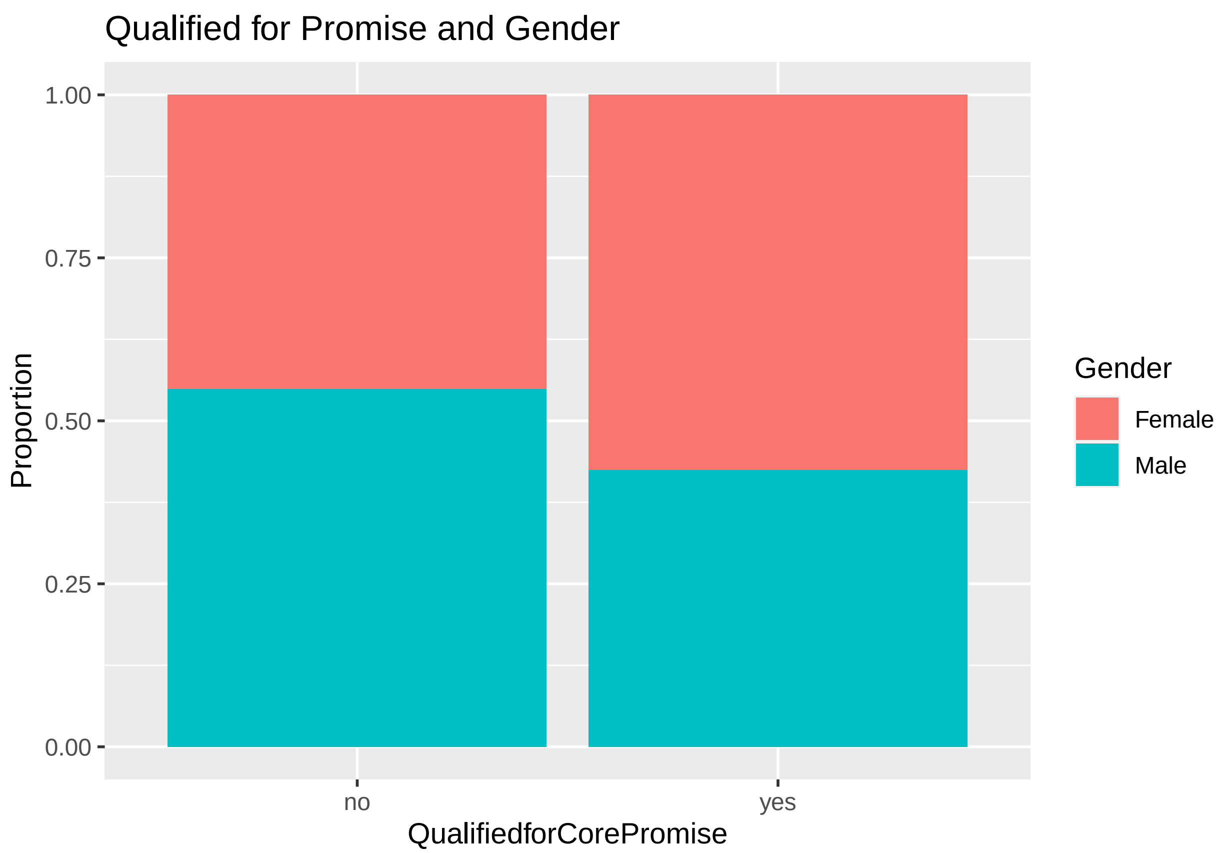


Figure 4: Receipt of Promise Scholarship vs. Gender

```
ggplot(qualified,
       aes(fill= Gender, x=EverReceivedPromiseAward)) +
  geom_bar(position="fill") +
  ggtitle("Ever Received Promise and Gender") +
  labs(y="Proportion")
```

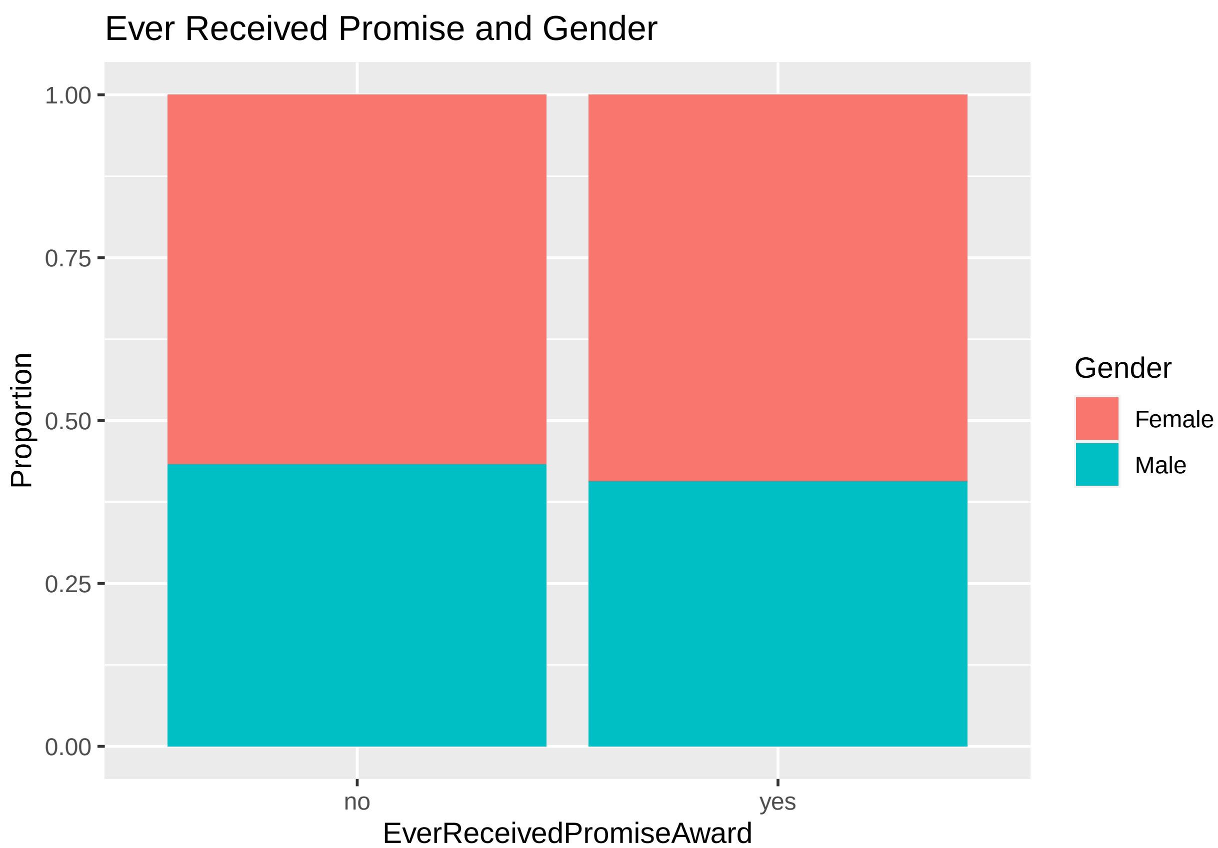


Figure 5: Receipt of Promise Scholarship vs. Career Certifications

```
ggplot(data = CTE_scholarship,
       aes(x = EverReceivedPromiseAward, y = num_cte)) +
  geom_boxplot(alpha = 0) +
  geom_jitter(alpha = 0.5, color = "green", width = 0.2, height = 0.2) +
  scale_y_continuous(breaks = c(0,2,4,6,8,10,12,14))+
  labs(title =
         "Boxplots of Number of Career Certifications Earned in terms of Whether Students Received Awar
       x = "Whether Students Received the Promise Scholarship",
       y = "Number of Certifications Earned")
```

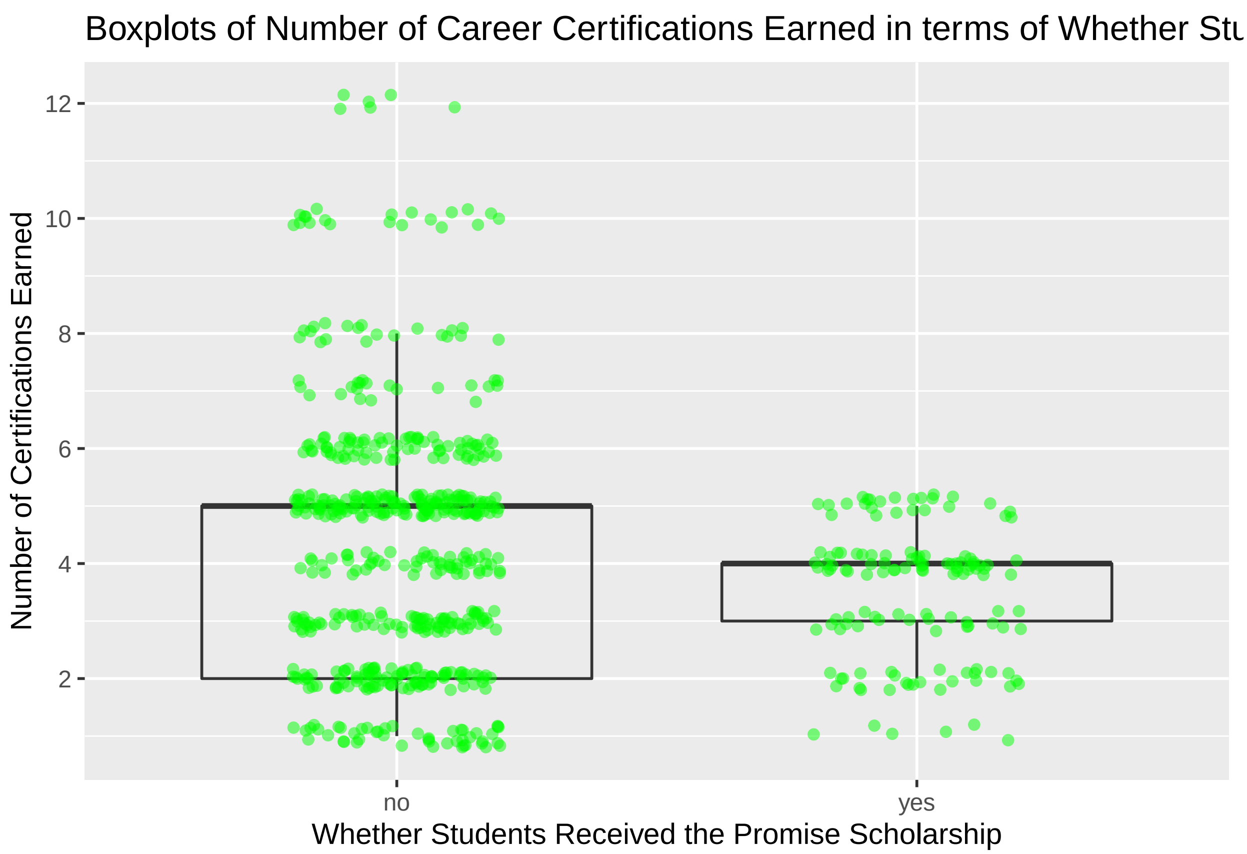


Figure 6: Receipt of Promise Scholarship vs. Economic Status

```
# Prepare data set for later visualization
join_data2c <-demo_scholarship_final %>%
  filter(QualifiedforCorePromise == "yes") %>%
  group_by(EconDisad,EverReceivedPromiseAward) %>%
  dplyr::summarise(counts = n())

ggplot(join_data2c, aes(x = EverReceivedPromiseAward, fill = EconDisad)) +
  geom_bar(position = "fill")+
  labs(title = "Barplots of Whether Received Promise under Different Economic Status",
       y = "Proportion")
```

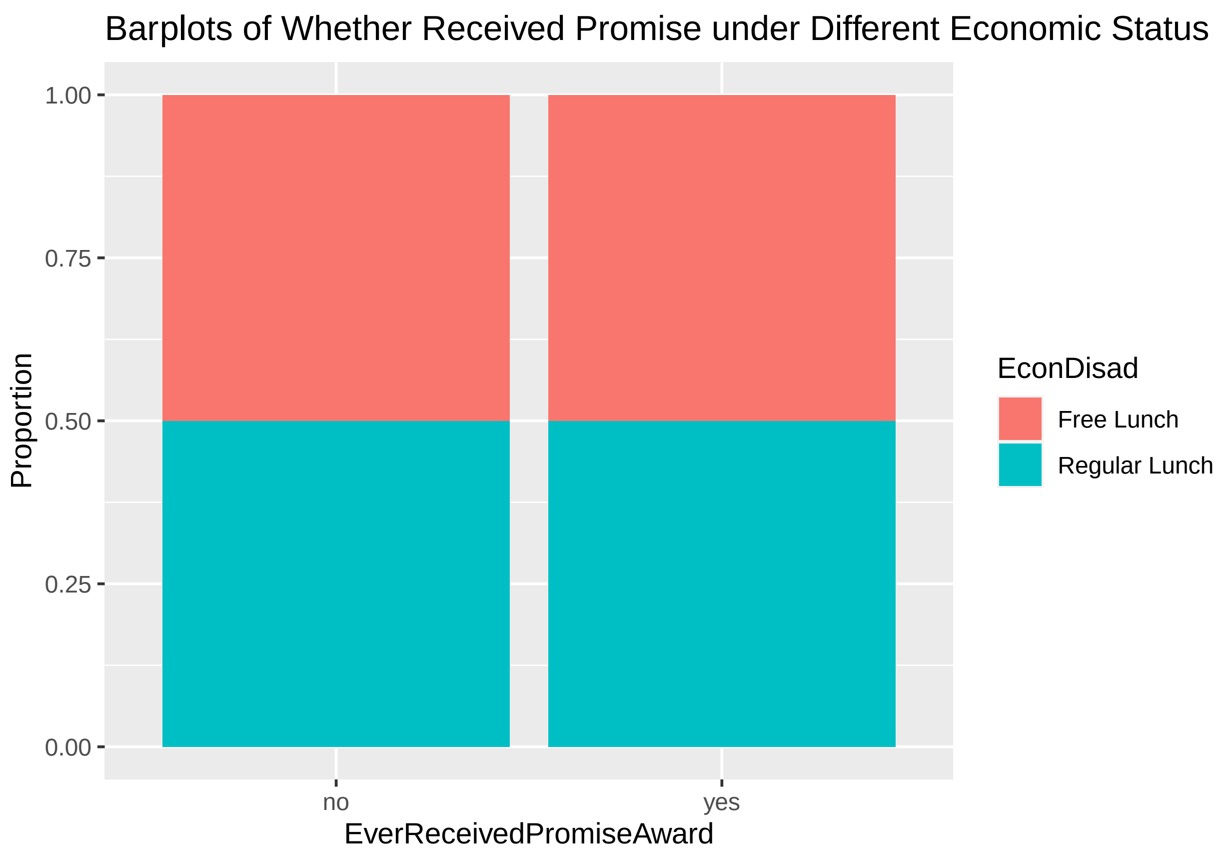


Figure 7: Cutoff vs. Qualification for Promise Scholarship

```r
# Prepare data set for later visualization
senior_gpa$RandomID <- as.character(senior_gpa$RandomID)
final_data <- attendance_rate %>%
  left_join(senior_gpa, by = c("RandomID"="RandomID"))
final_data_clean <- final_data %>%
  na.omit(final_data)

ggplot(data = final_data_clean,
       aes(x = excused_pct,
           y = CumulativeGPA,
           color = QualifiedforCorePromise)) +
  geom_point() +
  geom_hline(aes(yintercept=2.5), linetype="dashed") +
  geom_vline(aes(xintercept=0.9), linetype="dashed") +
  xlab("Attendance Rate") +
  ylab("Cumulative GPA") +
  ggtitle("GPA vs Attendance Rate") +
  guides(color = guide_legend("Qualified for Promise"))
```

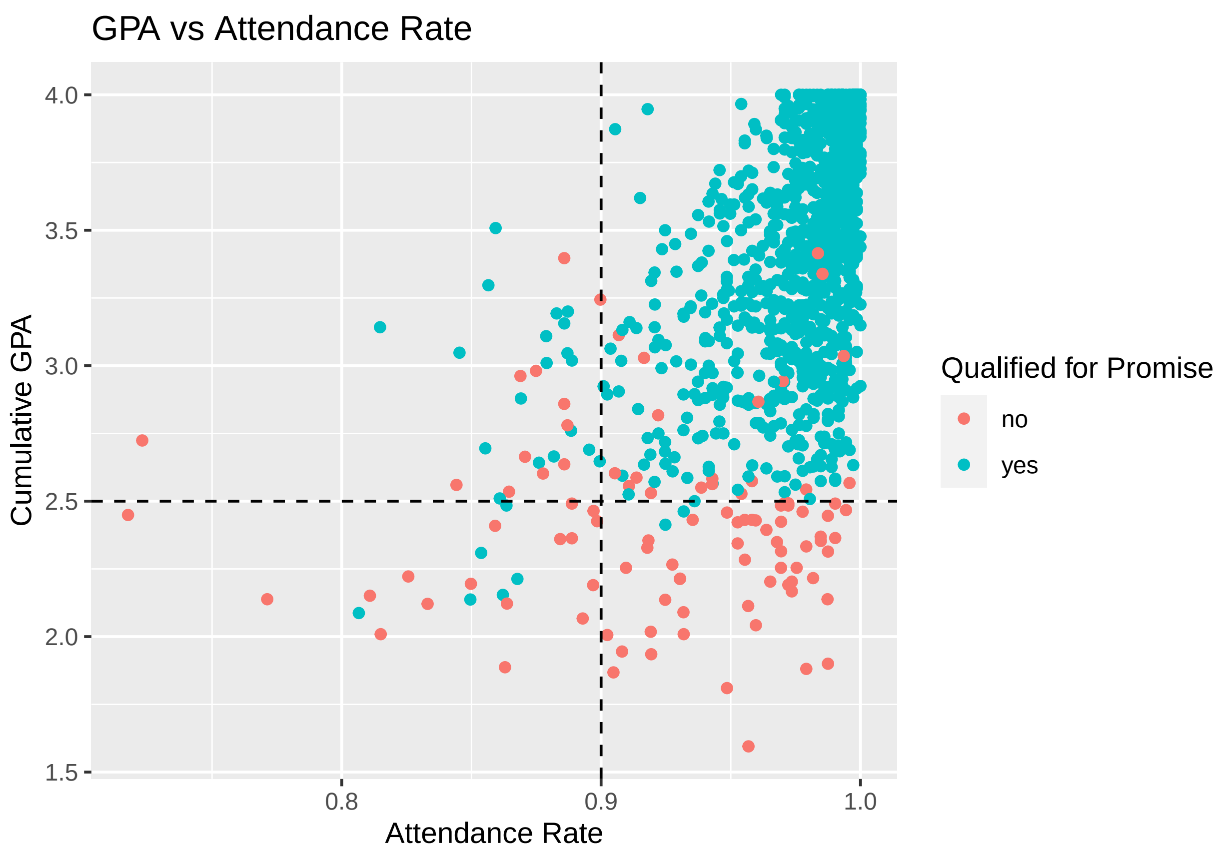
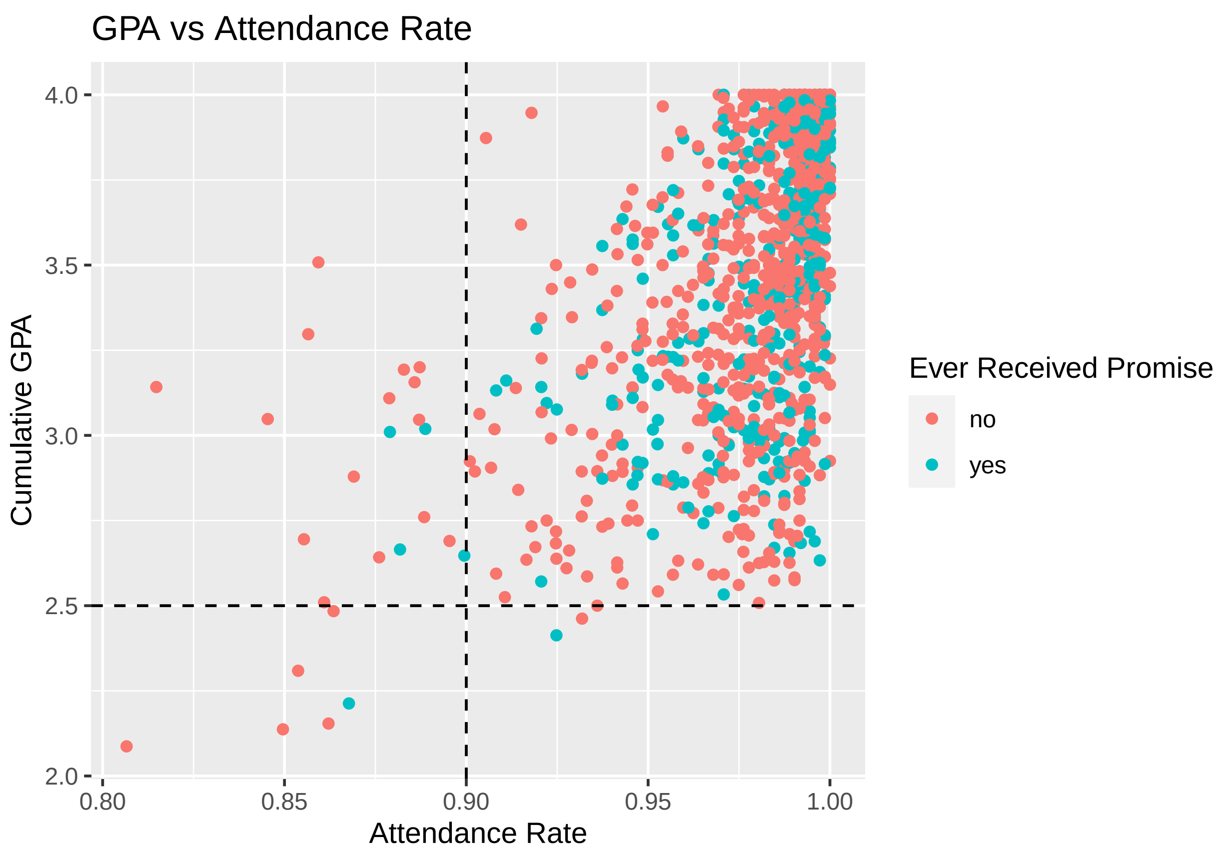


Figure 8: Cutoff vs. Receipt of Promise Scholarship

```r
# Filter qualified student in the data set
final_data_clean_qualifiedYes <- final_data_clean %>%
  filter(QualifiedforCorePromise == "yes")

ggplot(data = final_data_clean_qualifiedYes,
       aes(x = excused_pct,
           y = CumulativeGPA,
           color = EverReceivedPromiseAward)) +
  geom_point() +
  geom_hline(aes(yintercept=2.5), linetype="dashed") +
  geom_vline(aes(xintercept=0.9), linetype="dashed") +
  xlab("Attendance Rate") +
  ylab("Cumulative GPA") +
  ggtitle("GPA vs Attendance Rate") +
  guides(color = guide_legend("Ever Received Promise"))
```


GPA vs Attendance Rate
Cumulative GPA
Attendance Rate
4.0
3.5
3.0
2.5
2.0
0.80
0.85
0.90
0.95
1.00
Ever Received Promise
no
yes

Technical Appendix B: Scholarship Analysis

Data Wrangling

```r
# Library packages
library(tidyverse)
library(dplyr)
library(visdat)
library(stats)
library(ggplot2)
library(DHARMa)
library(arm)
library(readxl)
```

```r
# Import data
scholarship <- read_excel("DR-826 Pittsburgh Promise scholarship use.xlsx", sheet = "Sheet1")
NSC <- read_excel("DR-844 NSC data_not including 1920.xlsx", sheet = "NSC data")
sat <- read_excel("DR-834 SAT scores.xlsx", sheet = "SAT Scores")
ap <- read_excel("DR-834 AP scores.xlsx", sheet = "AP Exam Scores")
keystone <- read_excel("DR-836 Keystone assessment results.xlsx", sheet = "Keystone data")
enrollment <- read_excel("DR-828 Enrollment.xlsx", sheet = "Enrollment records")
```

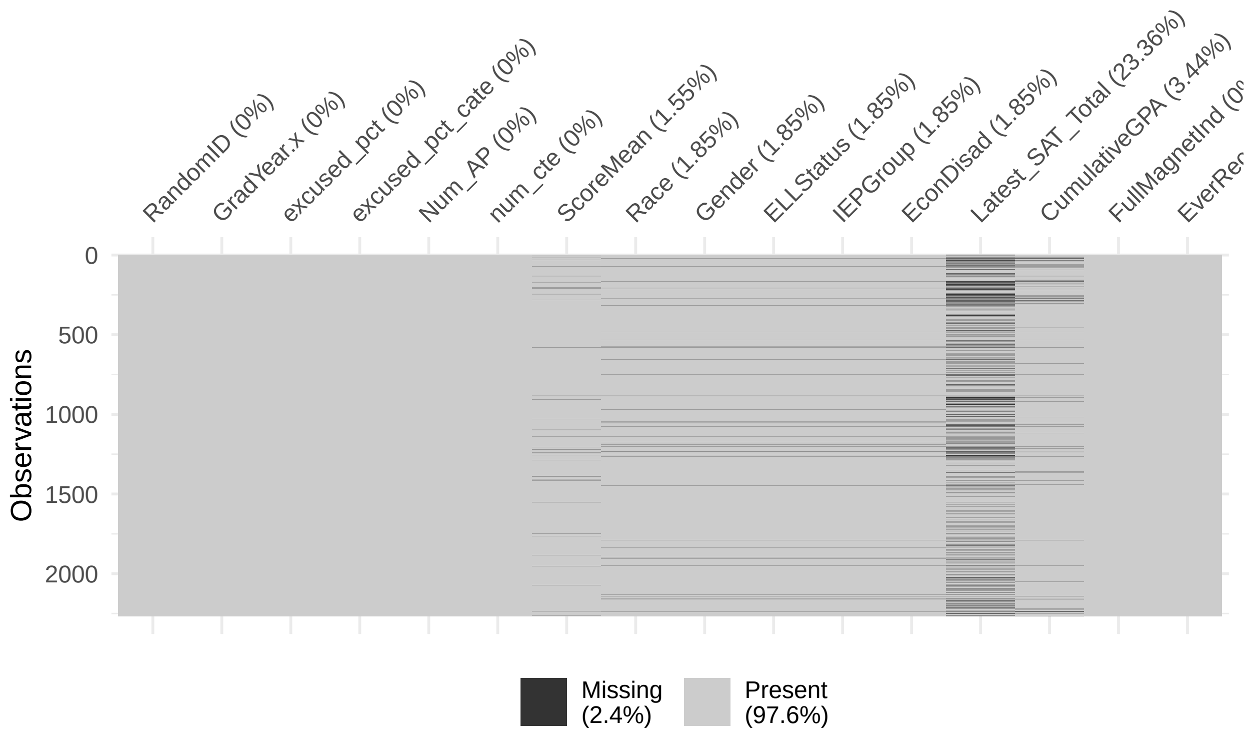
```r
# Clean and reformat the data above
# Create new data sets for data compilation later

# Create magnet_model data
# Serve as indicator of whether students attend magnet school
enrollment_new <- enrollment %>%
  mutate_at(c(2:14), as.factor) %>%
  mutate(EntryDate=as.Date(as.character(EntryDate), tryFormats = c("%Y%m%d"))) %>%
  mutate(WithdrawDate=as.Date(as.character(WithdrawDate), tryFormats = c("%Y%m%d")))

magnet_model <- enrollment_new %>%
  filter(FullMagnetInd == "1") %>%
  distinct(RandomID, EnrolledSchoolID, SchoolReportName, FullMagnetInd)
magnet_model$RandomID <- as.character(magnet_model$RandomID)

# Create keystone_model data
# Provide information about students' keystone scores
keystone_new <- keystone %>%
  mutate_at(c(1,3:7), as.factor) %>%
  mutate(AdminScaleScore=as.numeric(AdminScaleScore))

keystone_model <- keystone_new %>%
  na.omit() %>%
```

```r
  group_by(RandomID)%>%
  dplyr::summarise(ScoreMean = mean(AdminScaleScore))
keystone_model$RandomID <- as.character(keystone_model$RandomID)

# Create sat_model data
# Provide information about each student's highest SAT score
sat_new <- sat %>%
  mutate_at(2, as.factor) %>%
  mutate(LATEST_ASSESSMENT_DATE=as.Date(LATEST_ASSESSMENT_DATE)) %>%
  mutate_at(c(4:6), as.numeric)

sat_rep <- sat_new$RandomID[duplicated(sat_new$RandomID)]

sat_model <- sat_new %>%
  distinct(RandomID, Latest_SAT_Total, .keep_all = TRUE)
sat_model$RandomID <- as.character(sat_model$RandomID)

# Create nsc_model data
# Joined data set between NSC and scholarship
nsc2 <- NSC %>%
  dplyr::select(-c("Cohort")) %>%
  mutate(GradYear = substr(as.character(HIGH_SCHOOL_GRAD_DATE),1,4)) %>%
  mutate(EnrollYear = substr(as.character(ENROLLMENT_BEGIN),1,4)) %>%
  filter(EnrollYear == GradYear)

nsc2$RandomID <- as.character(nsc2$RandomID)
scholarship$GradYear <- as.character(scholarship$GradYear)
names(scholarship)[1] <- "RandomID"

nsc_scholarship <- inner_join(nsc2, scholarship, by = c("RandomID", "GradYear"))

nsc_model <- nsc_scholarship %>%
  dplyr::select(-c("HIGH_SCHOOL_GRAD_DATE",
            "ENROLLMENT_BEGIN",
            "ENROLLMENT_END")) %>%
  distinct()
nsc_model$RandomID <- as.character(nsc_model$RandomID)
```

```r
# Import data sets generated before
# Please refer to Appendix A for more details in code

# attendance_rate is the joined data set between attendance and scholarship
attendance_model <- read.csv("attendance_rate.csv")
attendance_model$RandomID <- as.character(attendance_model$RandomID)

# cte_scholarship is the joined data set between CTE and scholarship
cte_model <- read.csv("cte_scholarship.csv")
cte_model$RandomID <- as.character(cte_model$RandomID)

# senior_gpa is the cumulative GPA data generated from the original GPA data
senior_gpa <- read.csv("senior_gpa.csv")
senior_gpa$RandomID <- as.character(senior_gpa$RandomID)
```

```r
# demo_scholarship_final is the joined data set between demographics and scholarship
demographics_model <- read.csv("demo_scholarship_final.csv")
demographics_model$RandomID <- as.character(demographics_model$RandomID)

# ap_scholarship is the joined data set between ap and scholarship
ap_model <- read.csv("ap_scholarship.csv")
ap_model$RandomID <- as.character(ap_model$RandomID)
```

```r
# Data cleaning and reformatting
# Left join all data sets
data_join <- attendance_model %>%
  left_join(ap_model, by = c("RandomID"="RandomID")) %>%
  left_join(cte_model, by = c("RandomID"="RandomID")) %>%
  left_join(keystone_model, by = c("RandomID"="RandomID")) %>%
  left_join(demographics_model, by = c("RandomID"="RandomID")) %>%
  left_join(sat_model, by = c("RandomID"="RandomID")) %>%
  left_join(senior_gpa, by = c("RandomID"="RandomID")) %>%
  left_join(magnet_model, by = c("RandomID"="RandomID"))

# Select variables by column names
# Delete duplicated rows
data_variables <- data_join %>%
  dplyr::select(RandomID, GradYear.x, excused_pct, excused_pct_cate, Num_AP, num_cte,
         ScoreMean, Race, Gender, ELLStatus, IEPGroup, EconDisad, Latest_SAT_Total,
         CumulativeGPA, FullMagnetInd, EverReceivedPromiseAward.x) %>%
  distinct()

# Deal with NAs
# Replace missing num_AP, num_cte, FullMagnetInd with 0
data_na <- data_variables %>%
  replace_na(list(Num_AP=0, num_cte=0, FullMagnetInd=0))

# Visualize NAs
vis_miss(data_na)
```




```r
# Omit all NAs
data_omit <- data_na %>%
  na.omit()

# Change response variable to 0,1
data_omit$EverReceivedPromiseAward.x <- as.numeric(as.factor(data_omit$EverReceivedPromiseAward.x))-1

# Change variables into appropriate format
data_everReceived <- data_omit %>%
  mutate_at(c(2, 4,8:12,15:16), as.factor)

# Rename columns
colnames(data_everReceived)<- c("RandomID","GradYear","AttendanceRate", "AttendaceRateCate",
                                "Num_AP", "Num_CTE", "KeystoneMean", "Race", "Gender",
                                "ELLStatus", "IEPGroup", "EconDisad", "SAT_Total",
                                "CumulativeGPA","MagnetInd", "EverReceivedPromiseAward")
data_everReceived$Race_new <- ifelse(data_everReceived$Race != "White" &
                                       data_everReceived$Race != "African American", "Others",
                                     as.character(data_everReceived$Race))
data_everReceived$Race_new <- as.factor(data_everReceived$Race_new)
```

Part I: Logistic Regression Analysis for All Students

```r
# Construct null model and full model
model_null <- glm(EverReceivedPromiseAward ~ Race_new+Gender, data = data_everReceived,
                  family = "binomial")
model_full <- glm(EverReceivedPromiseAward ~ AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race_new
              +Gender+ELLStatus+IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd,
              data = data_everReceived, family = "binomial")
```

```r
# Stepwise selection on AIC
# Backwards selection
backwards <- step(model_full, scope=list(lower=formula(model_null),upper=formula(model_full)),
                  direction="backward", trace = 0)

# Forwards selection
forwards <- step(model_null, scope=list(lower=formula(model_null), upper=formula(model_full)),
                 direction="forward", trace = 0)

# Selection on both Directions
bothways <- step(model_null, list(lower=formula(model_null), upper=formula(model_full)),
                 direction="both", trace=0)

# Summary of the selected models
# Choose the selected model on both directions as our final model
summary(bothways)
```

```
##
## Call:
## glm(formula = EverReceivedPromiseAward ~ Race_new + Gender +
##     CumulativeGPA + AttendanceRate + KeystoneMean + ELLStatus +
##     MagnetInd, family = "binomial", data = data_everReceived)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2897  -0.8882  -0.6592   1.2784   2.3905
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -3.725100   2.868124  -1.299 0.194015
## Race_newOthers      0.055956   0.188480   0.297 0.766559
## Race_newWhite      -0.266577   0.146155  -1.824 0.068163 .
## GenderMale         -0.091442   0.117153  -0.781 0.435075
## CumulativeGPA       0.903230   0.158258   5.707 1.15e-08 ***
## AttendanceRate      8.051132   2.027035   3.972 7.13e-05 ***
## KeystoneMean       -0.005891   0.001681  -3.504 0.000458 ***
## ELLStatusNot in ELL  1.100948   0.437938   2.514 0.011939 *
## MagnetInd1          0.248580   0.115826   2.146 0.031862 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 2040.0  on 1707  degrees of freedom
## Residual deviance: 1919.4  on 1699  degrees of freedom
## AIC: 1937.4
##
## Number of Fisher Scoring iterations: 4
```

```r
formula(backwards)
```

```
## EverReceivedPromiseAward ~ AttendanceRate + KeystoneMean + Race_new +
##     Gender + ELLStatus + CumulativeGPA + MagnetInd
```

```r
formula(forwards)
```

```
## EverReceivedPromiseAward ~ Race_new + Gender + CumulativeGPA +
##     AttendanceRate + KeystoneMean + ELLStatus + MagnetInd
```

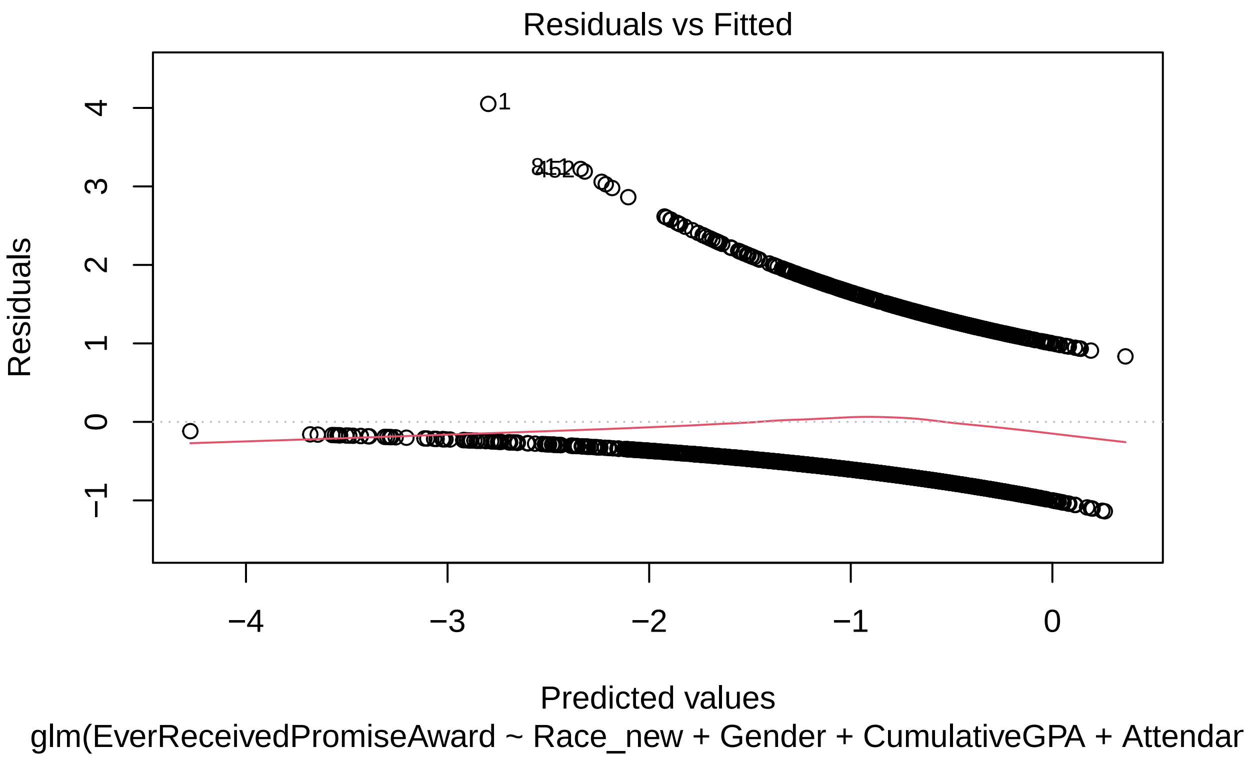
```r
formula(bothways)
```

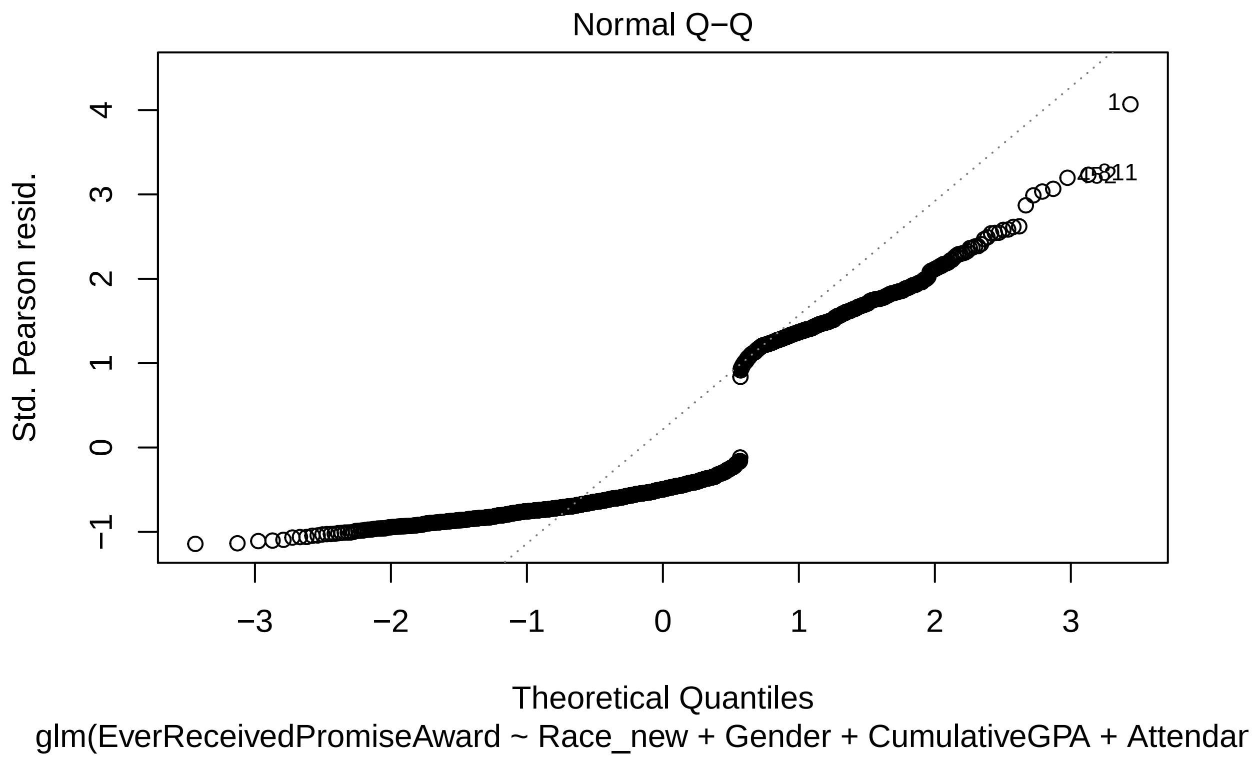
```
## EverReceivedPromiseAward ~ Race_new + Gender + CumulativeGPA +
##     AttendanceRate + KeystoneMean + ELLStatus + MagnetInd
```

Interpretations for all significant variables:

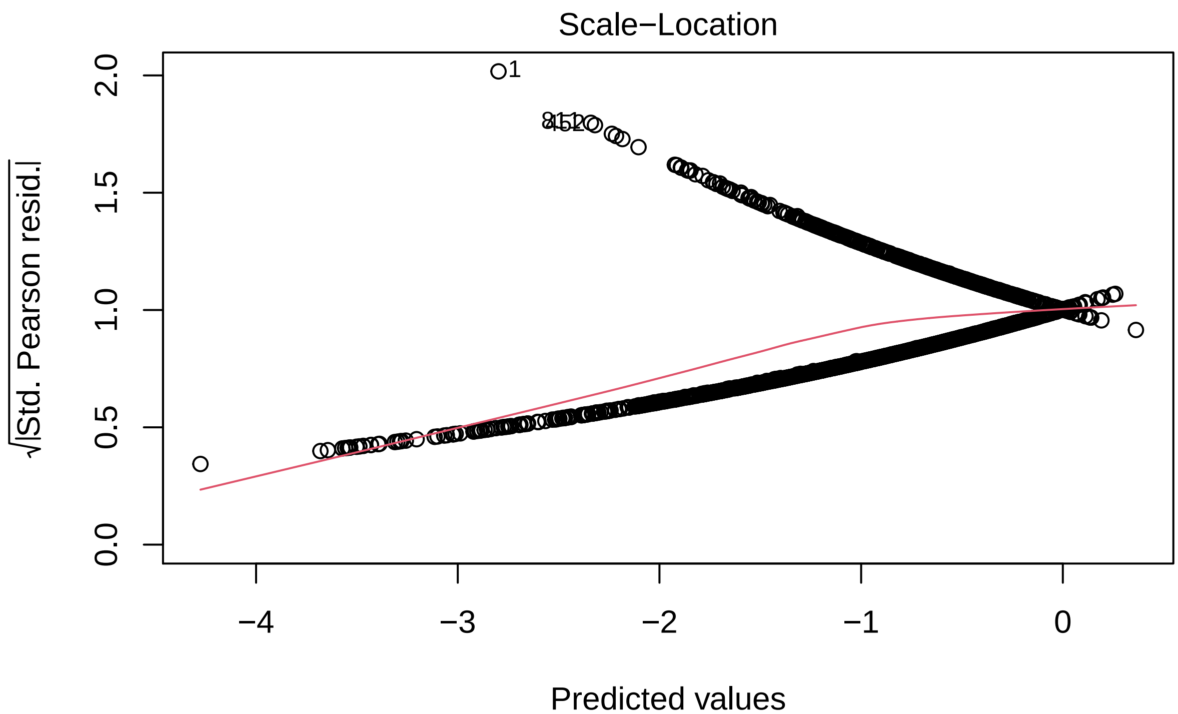
- When holding everything else fixed:
  - the odds to receive Promise for white students are 30.6% lower than black students;
  - the odds to receive Promise for student who are in magnet school are 38.3% higher than those in non-magnet school;
  - the odds for student who are not in ELL group are 200.7% higher than those in ELL group;
  - for 0.1 increase in GPA, we expect 9.4% increase in the odds of receiving Promise;
  - for 0.01 increase in attendance rate, we expect 8.4% increase in the odds of receiving Promise;
  - for 100 increase in Keystone mean, we expect 55.5% decrease in the odds of receiving Promise.

```r
# Model diagnostics
plot(bothways)
```

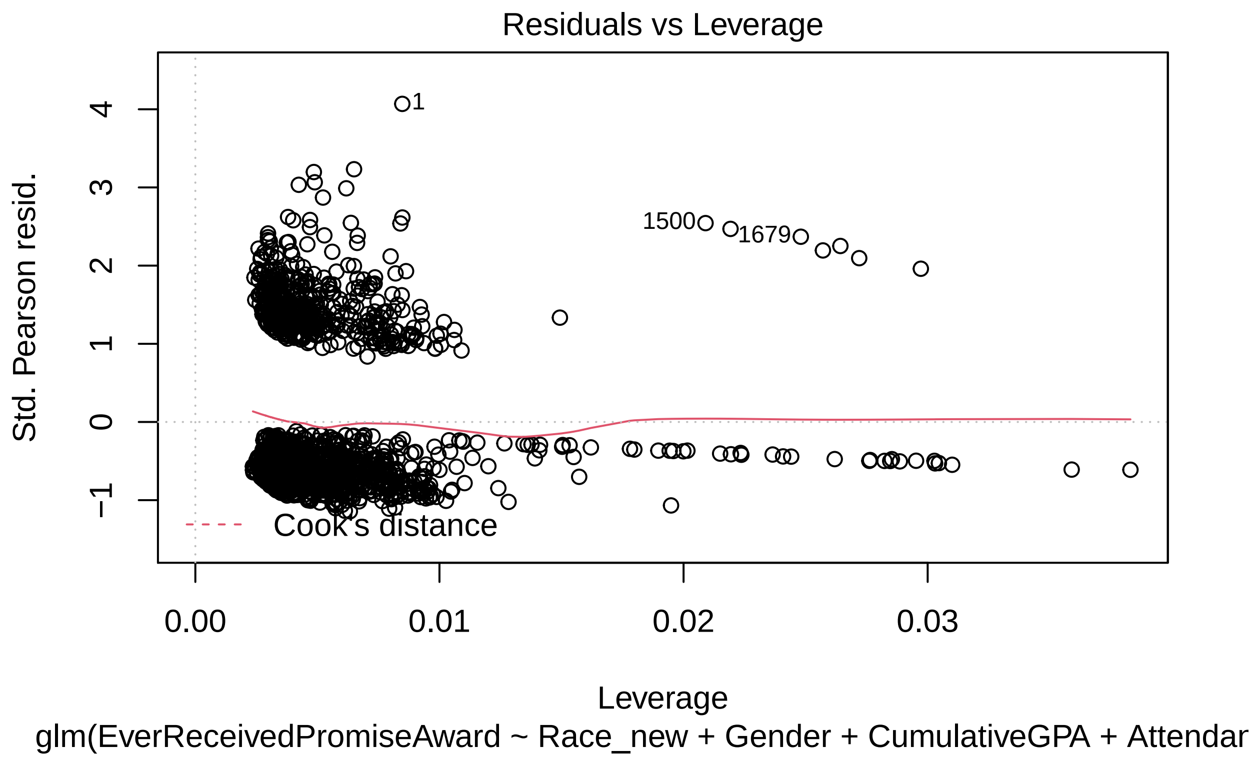

Residuals vs Fitted
Residuals
Predicted values
glm(EverReceivedPromiseAward ~ Race_new + Gender + CumulativeGPA + Attendar
4
3
2
1
0
−1
−4
−3
−2
−1
0
1
811
452



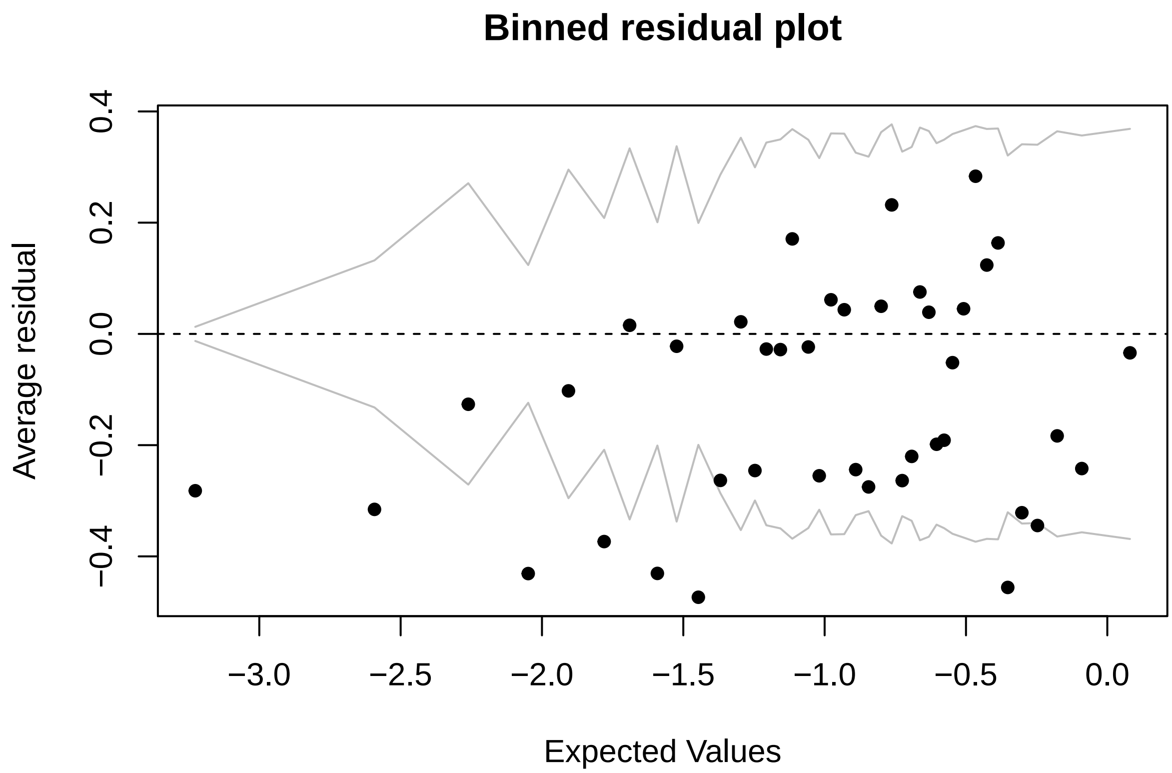
Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(EverReceivedPromiseAward ~ Race_new + Gender + CumulativeGPA + Attendar



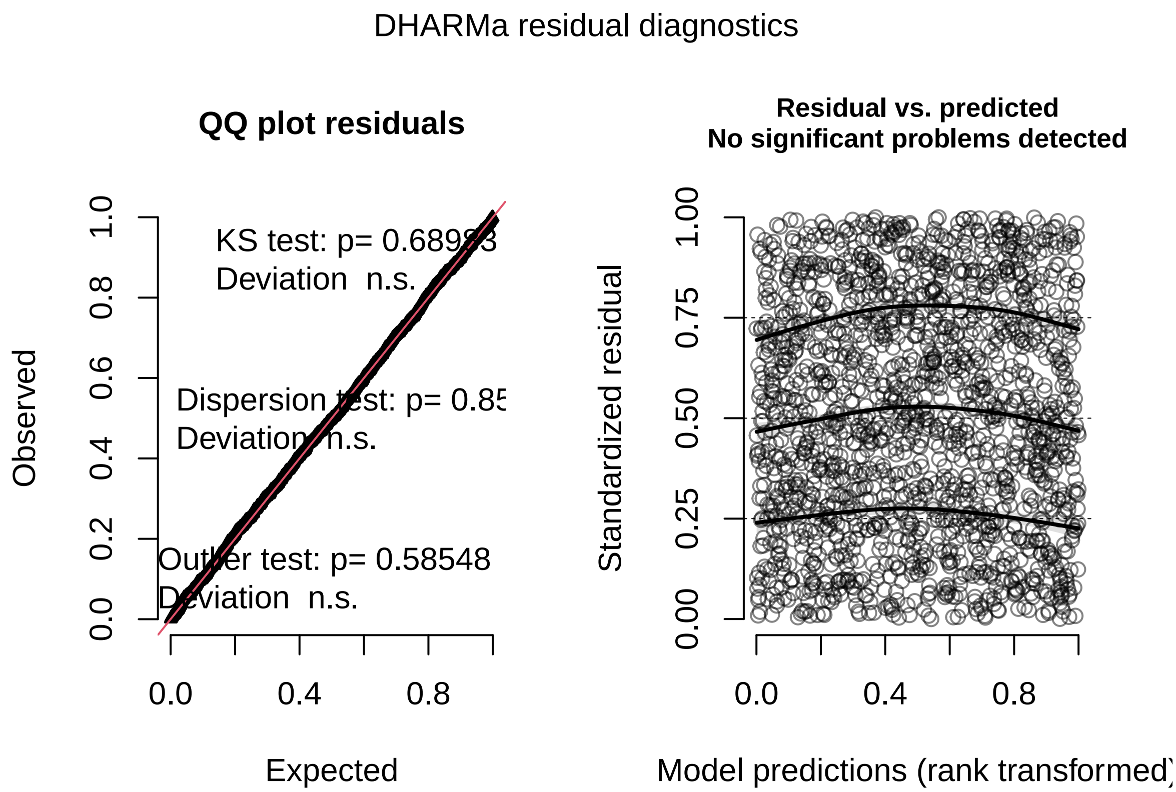
glm(EverReceivedPromiseAward ~ Race_new + Gender + CumulativeGPA + Attendar




```
y <- resid(bothways)
x <- predict(bothways)
binnedplot(x,y)
```




```
sim <- simulateResiduals(bothways)
plot(sim)
```



Conclusion of model diagnostics:

- From the binned plot, we know that the selected model is a good fit.
  - gray lines refer to the 95% confidence interval;
  - most points fall within the confidence interval;
  - residuals are randomly scattered with no obvious pattern.
- From QQ residuals plot, the residuals follow normal distribution.
- From residuals vs predicted plot, variance assumption of residuals is not violated by the selected model.

Part II: Logistic Regression Analysis for Qualified Students

```
# Prepare data sets for the analysis on only qualified students
data_qualified <- merge(data_everReceived,
                        attendance_model[, c("RandomID", "QualifiedforCorePromise")], by="RandomID")

data_qualifiedYes <- data_qualified %>%
  filter(QualifiedforCorePromise == "yes")

data_qualifiedYes$Race_new <- ifelse(data_qualifiedYes$Race != "White" & data_qualifiedYes$Race
                                     != "African American", "Other",
                                     as.character(data_qualifiedYes$Race))
data_qualifiedYes$Race_new <- as.factor(data_qualifiedYes$Race_new)
data_qualifiedYes$QualifiedforCorePromise <- as.factor(data_qualifiedYes$QualifiedforCorePromise)
```

```
# Construct null model and full model
model_null <- glm(EverReceivedPromiseAward ~ Race_new+Gender,
                  data = data_qualifiedYes, family = "binomial")
model_full <- glm(EverReceivedPromiseAward ~ AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race_new
              +Gender+ELLStatus+IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd,
              data = data_qualifiedYes, family = "binomial")
```

```
# Stepwise selection on AIC
# Backwards selection
backwards <- step(model_full, scope=list(lower=formula(model_null), upper=formula(model_full)),
                  direction="backward", trace = 0)

# Forward selection
forwards <- step(model_null, scope=list(lower=formula(model_null),upper=formula(model_full)),
                 direction="forward", trace = 0)

# Selection on both directions
bothways <- step(model_null, list(lower=formula(model_null),upper=formula(model_full)),
                 direction="both", trace=0)

# Summary of selected model
# Choose the selected model on both directions as our final model
summary(bothways)
```

```
##
## Call:
## glm(formula = EverReceivedPromiseAward ~ Race_new + Gender +
##     AttendanceRate + MagnetInd + KeystoneMean + CumulativeGPA +
##     ELLStatus, family = "binomial", data = data_qualifiedYes)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2264  -0.9362  -0.7954   1.3551   2.0967
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -4.916612   3.335304  -1.474  0.14045
```

```
## Race_newOther        0.106687   0.197056   0.541  0.58823
## Race_newWhite       -0.204815   0.154436  -1.326  0.18477
## GenderMale          -0.022979   0.123329  -0.186  0.85219
## AttendanceRate      10.401616   2.575524   4.039 5.38e-05 ***
## MagnetInd1           0.225356   0.122491   1.840  0.06580 .
## KeystoneMean        -0.005469   0.001796  -3.044  0.00233 **
## CumulativeGPA        0.475464   0.195891   2.427  0.01522 *
## ELLStatusNot in ELL  0.783758   0.462010   1.696  0.08981 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1722.8  on 1356  degrees of freedom
## Residual deviance: 1675.7  on 1348  degrees of freedom
## AIC: 1693.7
##
## Number of Fisher Scoring iterations: 4
```

```r
formula(backwards)
```

```
## EverReceivedPromiseAward ~ AttendanceRate + KeystoneMean + Race_new +
##     Gender + ELLStatus + CumulativeGPA + MagnetInd
```

```r
formula(forwards)
```

```
## EverReceivedPromiseAward ~ Race_new + Gender + AttendanceRate +
##     MagnetInd + KeystoneMean + CumulativeGPA + ELLStatus
```

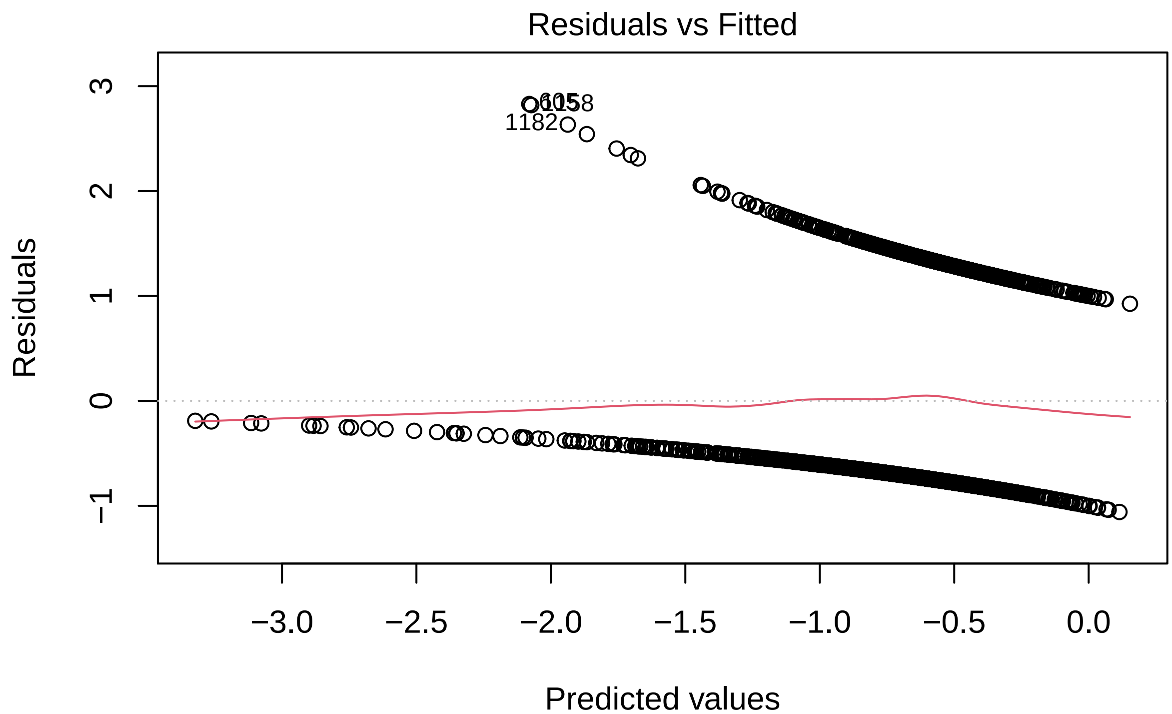
```r
formula(bothways)
```

```
## EverReceivedPromiseAward ~ Race_new + Gender + AttendanceRate +
##     MagnetInd + KeystoneMean + CumulativeGPA + ELLStatus
```

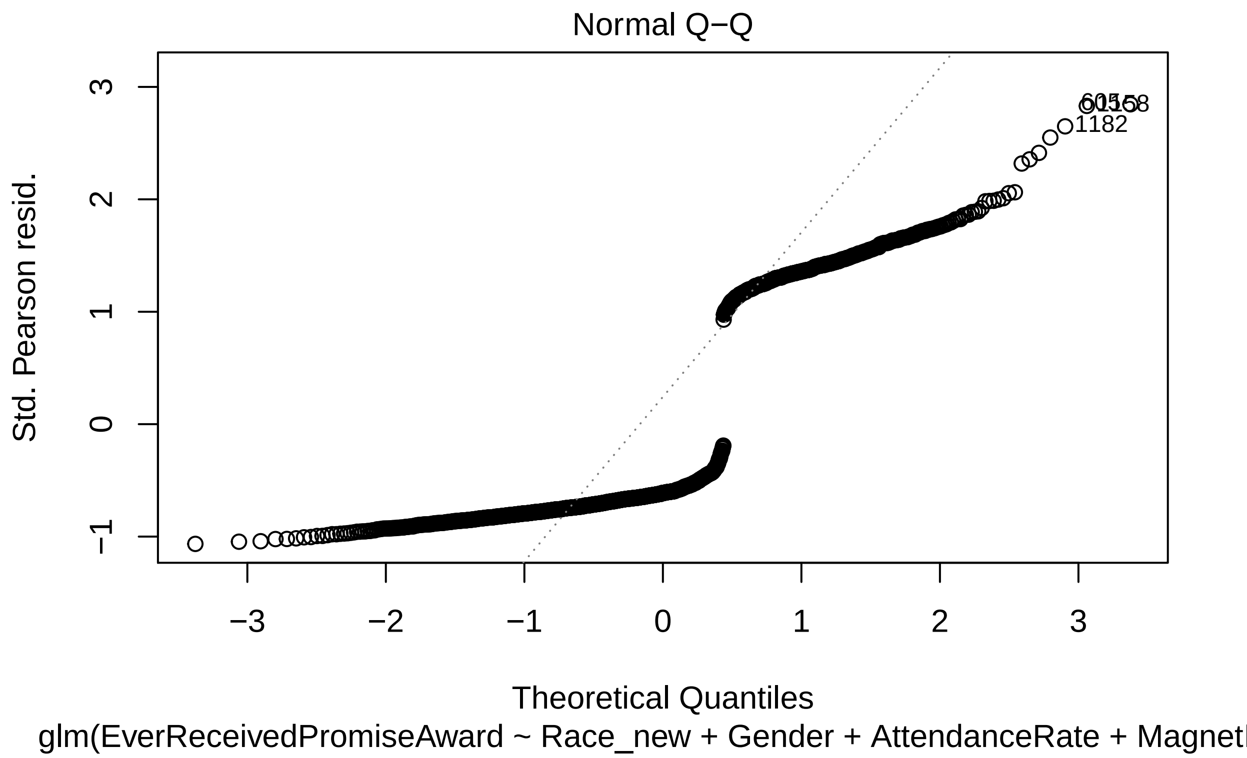
Interpretation for all significant variables:

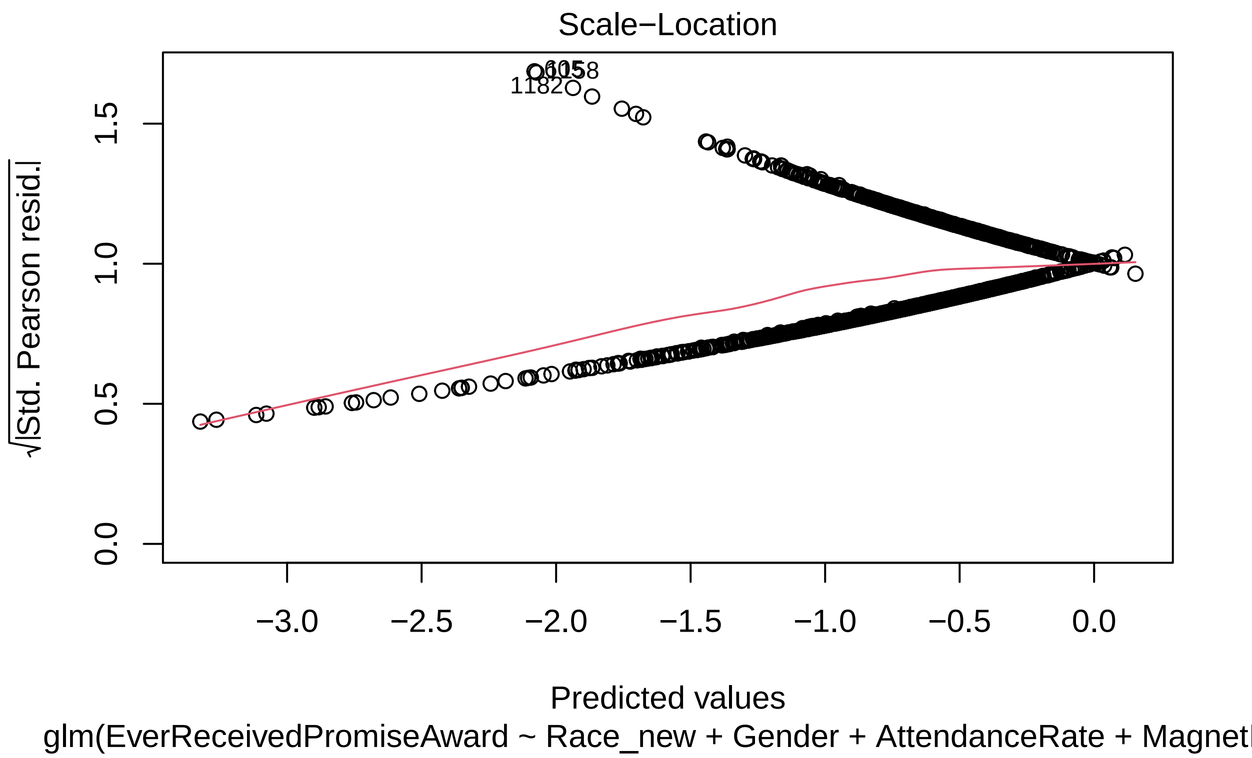
- When holding everything else fixed:
  - the odds to receive Promise for student who are in magnet school are 25.3% higher than those in non-magnet school;
  - the odds for student who are not in ELL group are 119% higher than those in ELL group;
  - for 0.1 increase in GPA, we expect 4.9% increase in the odds of receiving Promise;
  - for 0.01 increase in attendance rate, we expect 11% increase in the odds of receiving Promise;
  - for 100 increase in Keystone mean, we expect 57.9% decrease in the odds of receiving Promise.

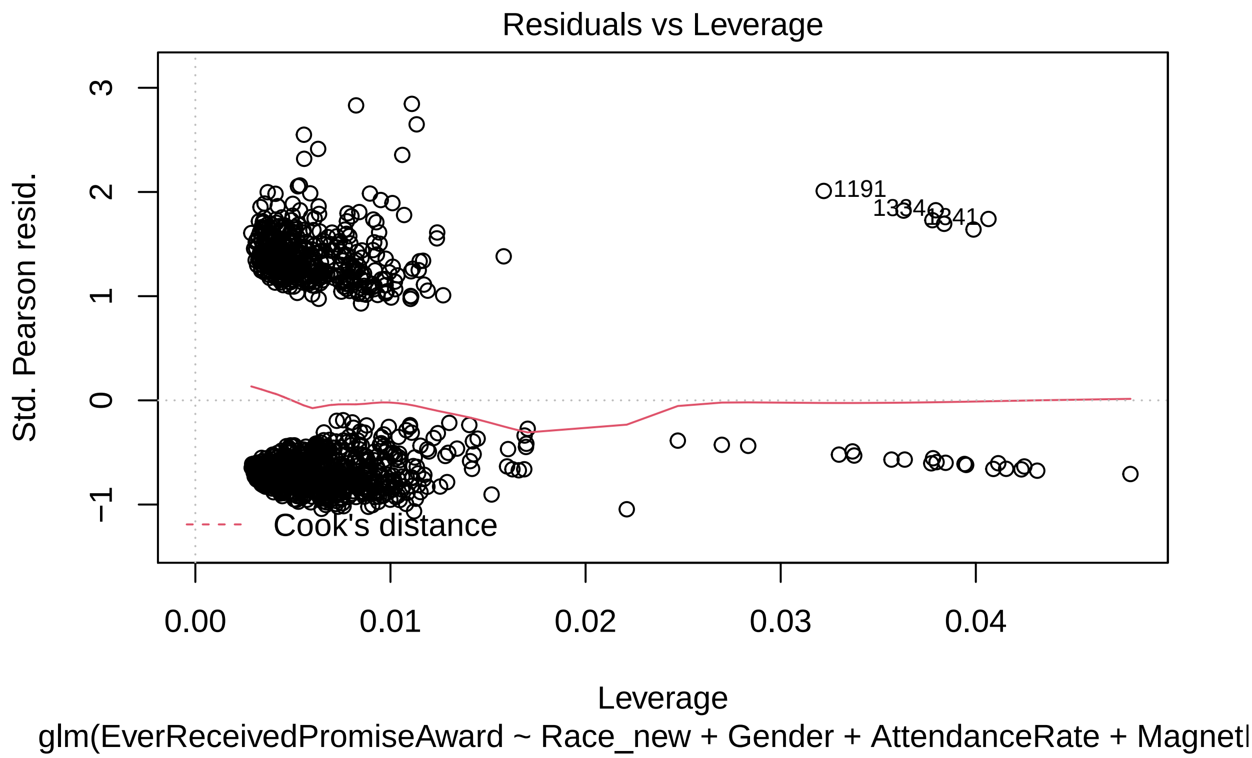
```r
# Model diagnostics
plot(bothways)
```



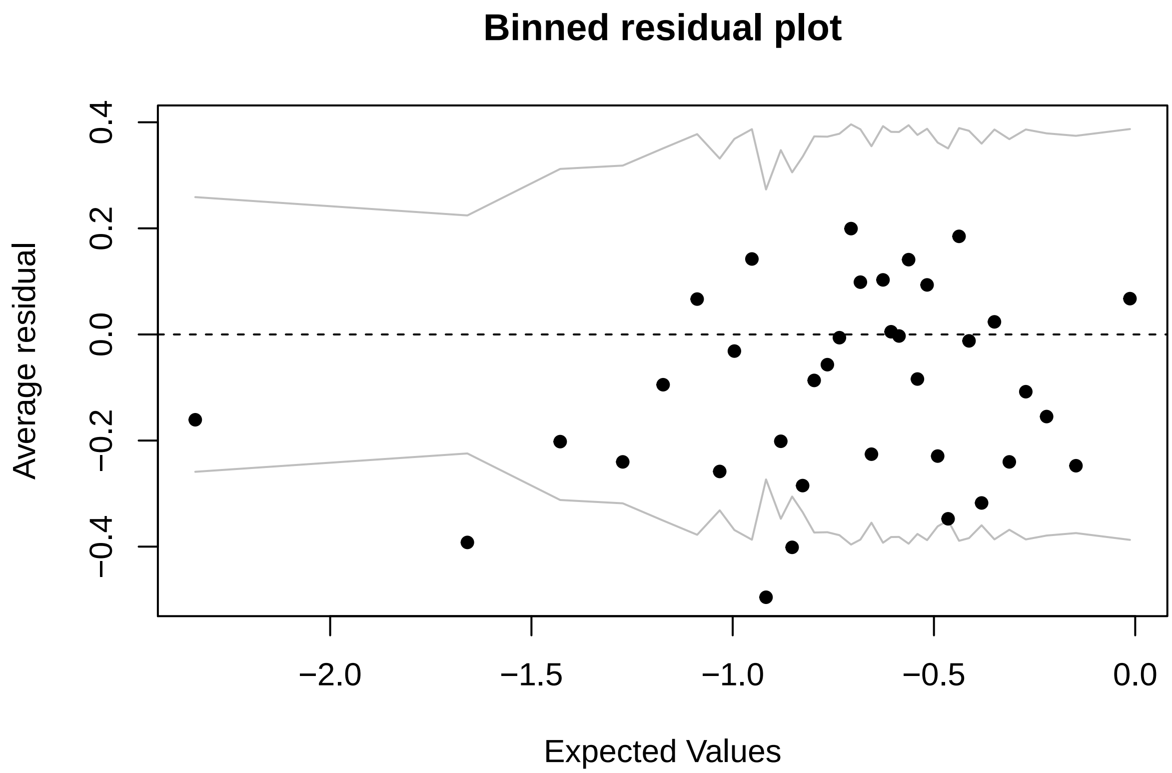
glm(EverReceivedPromiseAward ~ Race_new + Gender + AttendanceRate + MagnetI


Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(EverReceivedPromiseAward ~ Race_new + Gender + AttendanceRate + MagnetI
1182
3
2
1
0
−1
−3
−2
−1
0
1
2
3

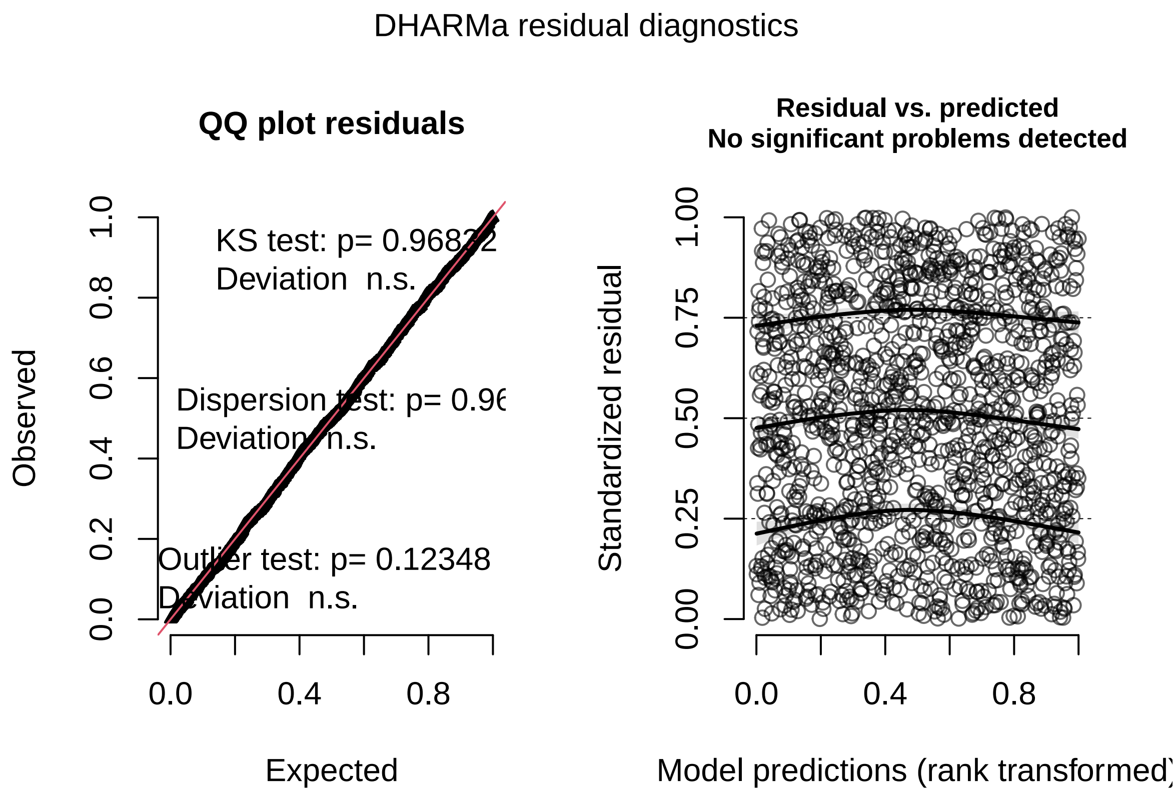

Scale−Location
605
6058
1182
1.5
1.0
0.5
0.0
√|Std. Pearson resid.|
−3.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
Predicted values
glm(EverReceivedPromiseAward ~ Race_new + Gender + AttendanceRate + Magnet




```r
y <- resid(bothways)
x <- predict(bothways)
binnedplot(x,y)
```




```r
sim <- simulateResiduals(bothways)
plot(sim)
```



Conclusion of model diagnostics:

- From the binned plot, we know that the selected model is a good fit.
  - gray lines refer to the 95% confidence interval;
  - most points fall within the confidence interval;
  - residuals are randomly scattered with no obvious pattern.
- From QQ residuals plot, the residuals follow normal distribution.
- From residuals vs predicted plot, variance assumption of residuals is not violated by the selected model.

Technical Appendix C: Retention Analysis

Part I: Exploratory Data Analysis & T-tests

Data Wrangling

```
# Import packages
library(plyr)
library(dplyr)
library(lubridate)
library(tidyverse)
library(readxl)
library(ggplot2)
library(arm)
```

```
# Import scholarship data
scholarship <- read_excel("DR-826 Pittsburgh Promise scholarship use.xlsx", sheet = "Sheet1")

# Clean scholarship data
scholarship <- scholarship %>%
  mutate(QualifiedforCorePromise = ifelse(QualifiedforCorePromise == "yes",1,0),
         QualifiedforExtensionPromise = ifelse(QualifiedforExtensionPromise == "yes",1,0),
         EverReceivedPromiseAward = ifelse(EverReceivedPromiseAward == "yes",1,0),
         StillReceivingAward = ifelse(StillReceivingAward == "yes",1,0),
         StillEligible = ifelse(StillEligible == "yes",1,0),
         HighSchool = as.factor(HighSchool)) %>%
  rename(RandomID = "Random ID")
head(scholarship)
```

```
## # A tibble: 6 x 8
##   RandomID GradYear QualifiedforCor~ QualifiedforExt~ EverReceivedPro~
##   <chr>       <dbl>            <dbl>            <dbl>            <dbl>
## 1 5829765      2018                0                0                0
## 2 5832055      2018                0                0                0
## 3 5833420      2018                1                0                0
## 4 5840516      2018                0                0                0
## 5 5841218      2018                0                0                1
## 6 5847024      2018                1                0                0
## # ... with 3 more variables: StillReceivingAward <dbl>, StillEligible <dbl>,
## #   HighSchool <fct>
```

```
# Import NSC data
nsc <- read_excel("DR-844 NSC data_not including 1920.xlsx", sheet = "NSC data")
nsc$RandomID <- as.character(nsc$RandomID)
head(nsc)
```

```
## # A tibble: 6 x 11
##   RandomID Cohort HIGH_SCHOOL_GRA~ COLLEGE_STATE `2-YEAR_4-YEAR` PUBLIC_PRIVATE
##   <chr>     <dbl>            <dbl> <chr>         <chr>           <chr>
## 1 5841218    1415         20180608 PA            4-year          Public
## 2 5841218    1415         20180608 PA            4-year          Public
## 3 5847024    1415         20180608 NY            4-year          Private
## 4 5847024    1415         20180608 NY            4-year          Private
## 5 5847024    1415         20180608 NY            4-year          Private
## 6 5847024    1415         20180608 NY            4-year          Private
## # ... with 5 more variables: ENROLLMENT_BEGIN <dbl>, ENROLLMENT_END <dbl>,
## #   ENROLLMENT_STATUS <chr>, GRADUATED <chr>, GRADUATION_DATE <dbl>
```

```r
# Import demographics data
demo1415 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1415")
demo1516 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1516")
demo1617 <- read_excel("DR-827 Demographics.xlsx", sheet = "Demographics for cohort 1617")

# Clean demographics data
demographics <- rbind(demo1415,demo1516,demo1617)
demographics <- demographics %>%
  dplyr::select(RandomID, Race) %>%
  distinct()

# Delete students who have more than 1 races
demographics <- demographics %>%
  filter(! RandomID %in% c(6008133,6040930,
                           6126484,6137723,
                           6207255,6213261,
                           6228923,6510341))
```

Comparison I: Retention vs. Scholarship Receipt

EDA

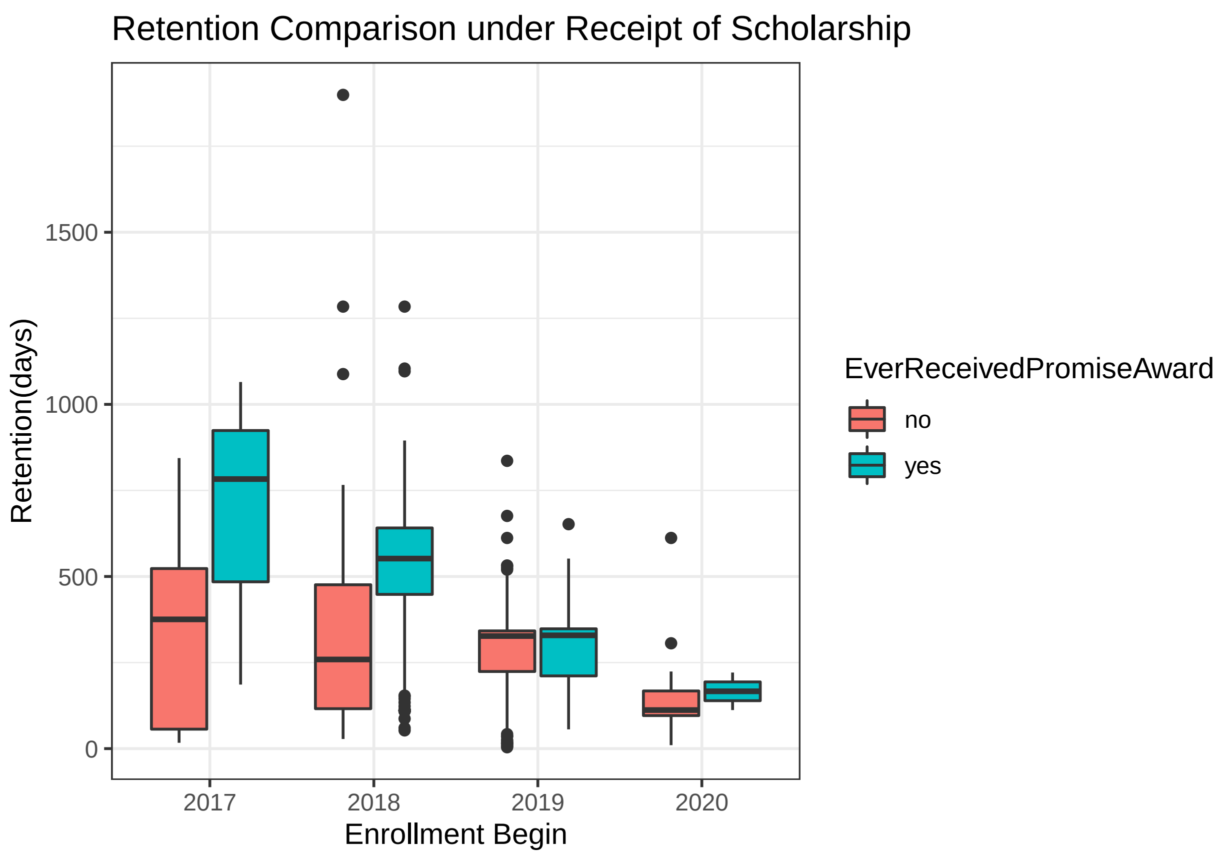
```r
# Prepare data sets for the visualization of comparison plot
nsc_processed = nsc %>%
  filter(ENROLLMENT_BEGIN >0) %>%
  filter(ENROLLMENT_BEGIN %/% 10000 != 2016) %>%
  filter(COLLEGE_STATE == "PA") %>%
  mutate(ENROLLMENT_BEGIN = as.Date(paste(substr(ENROLLMENT_BEGIN, 1, 4),
                                          substr(ENROLLMENT_BEGIN, 5, 6),
                                          substr(ENROLLMENT_BEGIN, 7, 8),
                                          sep = "-")),
         ENROLLMENT_END = as.Date(paste(substr(ENROLLMENT_END, 1, 4),
                                        substr(ENROLLMENT_END, 5, 6),
                                        substr(ENROLLMENT_END, 7, 8),
                                        sep = "-")),
         elapsed = ENROLLMENT_END - ENROLLMENT_BEGIN)
```

```r
# Issue: some students enroll in fall while others enroll in spring
# Create a new variable semester to indicate which semester students enroll in
```

```r
nsc_retention = nsc_processed %>%
  group_by(RandomID) %>%
  summarise(retention = sum(elapsed),
            ENROLLMENT_BEGIN_year = year(min(ENROLLMENT_BEGIN)),
            semester = ifelse(month(min(ENROLLMENT_BEGIN))>6, "Fall","Spring"))
```

```r
# Prepare data sets for the visualization of comparison plot
nsc_promise = nsc_retention %>%
  left_join(scholarship, by = "RandomID") %>%
  mutate(EverReceivedPromiseAward =
           ifelse(EverReceivedPromiseAward==1,"yes","no"),
         EverReceivedPromiseAward = replace_na(EverReceivedPromiseAward,"no"),
         EverReceivedPromiseAward = as.factor(EverReceivedPromiseAward),
         retention = as.numeric(retention)) %>%
  dplyr::select(-c(QualifiedforCorePromise,
            QualifiedforExtensionPromise,
            StillReceivingAward,
            StillEligible,HighSchool))
```

```r
# Visualization of retention comparison under receipt of scholarship
ggplot(nsc_promise, aes(x=as.factor(ENROLLMENT_BEGIN_year),
                        y=retention,
                        fill=EverReceivedPromiseAward)) +
  geom_boxplot() +
  labs(x = "Enrollment Begin",
       y = "Retention(days)",
       title = "Retention Comparison under Receipt of Scholarship") +
  theme_bw()
```



Findings:

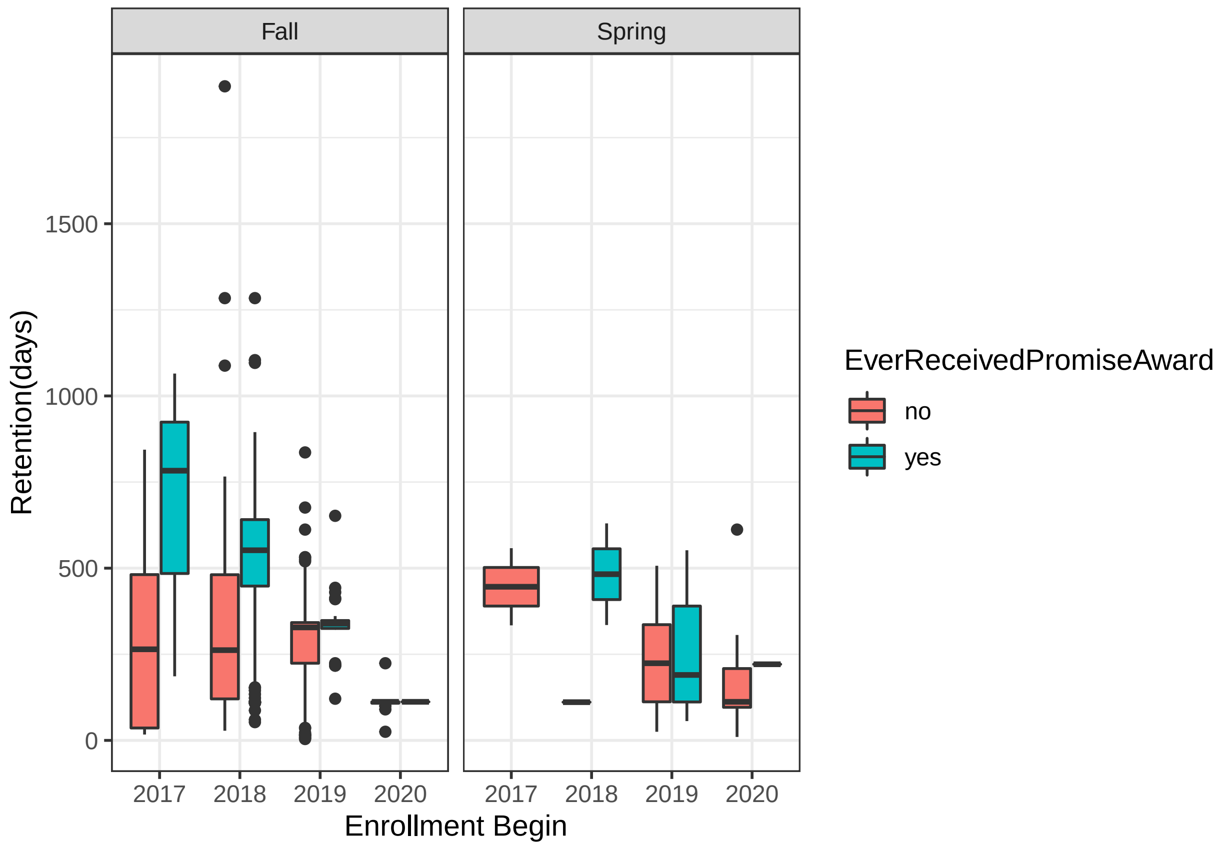
- We see that students who received scholarship tend to have better retention except for year 2019.
- However, notice that we only have 13 observations for year 2017, and 93 observations for year 2020.

```
# Take a closer look by semester they enroll
nsc_promise %>%
  group_by(ENROLLMENT_BEGIN_year, semester) %>%
  count()
```

```
## # A tibble: 8 x 3
## # Groups:   ENROLLMENT_BEGIN_year, semester [8]
##   ENROLLMENT_BEGIN_year semester     n
##                   <dbl> <chr>    <int>
## 1                  2017 Fall        11
## 2                  2017 Spring       2
## 3                  2018 Fall       571
## 4                  2018 Spring       3
## 5                  2019 Fall       639
## 6                  2019 Spring      59
## 7                  2020 Fall        28
## 8                  2020 Spring      65
```

```
# Visualization of retention comparison under receipt of scholarship (by semester)
# Since we have limited observations for 2017 and all spring semesters,
```

```r
# the resulting plot is just for reference
ggplot(nsc_promise, aes(x=as.factor(ENROLLMENT_BEGIN_year),
                        y=retention,
                        fill=EverReceivedPromiseAward)) +
  geom_boxplot() +
  labs(x = "Enrollment Begin",
       y = "Retention(days)") +
  theme_bw() +
  facet_wrap(~semester)
```



Check for Significance for 2018 & 2019 (T-tests)

```r
# Prepare data sets for 2018 & 2019
nsc_promise_2018 = nsc_promise %>% filter(ENROLLMENT_BEGIN_year == 2018)
nsc_promise_2019 = nsc_promise %>% filter(ENROLLMENT_BEGIN_year == 2019)

# Use Bartlett test to check for equal variance or not (2018)
bartlett.test(retention~EverReceivedPromiseAward, data = nsc_promise_2018) #unequal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by EverReceivedPromiseAward
## Bartlett's K-squared = 37.596, df = 1, p-value = 8.7e-10
```

```r
# T-test to check for significance of difference in retention (2018)
oneway.test(retention~EverReceivedPromiseAward, data = nsc_promise_2018, var.equal = FALSE)
```

```
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  retention and EverReceivedPromiseAward
## F = 44.755, num df = 1.00, denom df = 130.29, p-value = 5.98e-10
```

```r
# Use Bartlett test to check for equal variance or not (2019)
bartlett.test(retention~EverReceivedPromiseAward, data = nsc_promise_2019) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by EverReceivedPromiseAward
## Bartlett's K-squared = 2.1978, df = 1, p-value = 0.1382
```

```r
# T-test to check for significance of difference in retention (2019)
oneway.test(retention~EverReceivedPromiseAward, data = nsc_promise_2019, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  retention and EverReceivedPromiseAward
## F = 0.27045, num df = 1, denom df = 696, p-value = 0.6032
```

Findings:

- For 2018, the difference is significant; for 2019, the difference is not significant.
- Findings from significance tests align with what we observe in the box plot above.
- Thus, the difference in retention is quite obvious in 2018, whereas in 2019 is not very obvious. Maybe the retention difference would become more obvious for more senior students.

Comparison II: Retention vs.Race

EDA

```r
# Check racial composition
demographics %>% group_by(Race) %>% count() %>% arrange(-n)
```

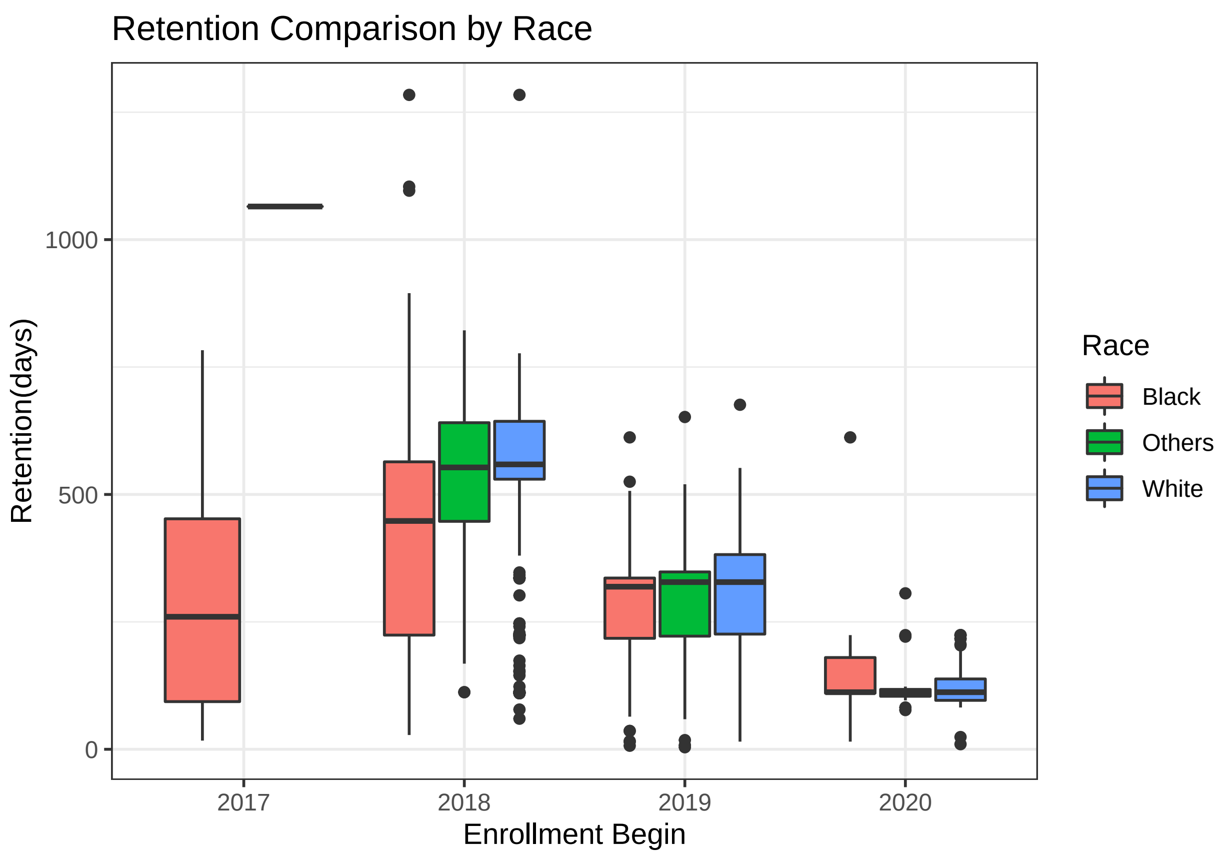
```
## # A tibble: 7 x 2
## # Groups:   Race [7]
##   Race                                  n
##   <chr>                             <int>
## 1 African American                   2766
## 2 White                              1847
## 3 Multi-Racial                        327
```

```
## 4 Asian (not Pacific Islander)                    180
## 5 Hispanic                                        164
## 6 American Indian                                  10
## 7 Native Hawaiian or other Pacific Islander         4
```

- We see that the majority of students are blacks and whites, so we will explore the difference in student retention between these two races and categorize the rest racial groups as "Other".
- Also, we restrict on students who went to college in PA.

```
# Prepare data sets for the visualization of the comparison plot
demographics$RandomID <- as.character(demographics$RandomID)
nsc_demographics = nsc_promise %>%
  left_join(demographics, by = "RandomID") %>%
  mutate(Race = ifelse(Race != "White" & Race != "African American",
                       "Others",
                       ifelse(Race == "African American",
                              "Black",
                              "White"))) %>%
  na.omit(retention)
```

```
# Visualization of retention comparison under race
ggplot(nsc_demographics, aes(x=as.factor(ENROLLMENT_BEGIN_year),
                             y=retention,
                             fill= Race)) +
  geom_boxplot() +
  labs(x = "Enrollment Begin",
       y = "Retention(days)",
       title = "Retention Comparison by Race") +
  theme_bw()
```



Check for Significance for 2018 & 2019 (T-tests)

```
# Prepare data set for the visualization of comparison plot
nsc_demographics2 = nsc_promise %>%
  left_join(demographics, by = "RandomID") %>%
  filter(Race %in% c("White","African American")) %>%
  mutate(Race = ifelse(Race == "White", "White","Black"))

# Prepare data sets for 2018 & 2019
nsc_demographics_2018 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2018)
nsc_demographics_2019 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2019)

# Use Bartlett test to check for equal variance or not (2018)
bartlett.test(retention~Race, data = nsc_demographics_2018) #unequal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 11.071, df = 1, p-value = 0.0008769
```

```r
# T-test to check for significance of difference in retention (2018)
oneway.test(retention~Race, data = nsc_demographics_2018, var.equal = FALSE)
```

```
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  retention and Race
## F = 20.256, num df = 1.00, denom df = 487.42, p-value = 8.48e-06
```

```r
# Use Bartlett test to check for equal variance or not (2019)
bartlett.test(retention~Race, data = nsc_demographics_2019) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 0.077868, df = 1, p-value = 0.7802
```

```r
# T-test to check for significance of difference in retention (2019)
oneway.test(retention~Race, data = nsc_demographics_2019, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  retention and Race
## F = 29.355, num df = 1, denom df = 591, p-value = 8.782e-08
```

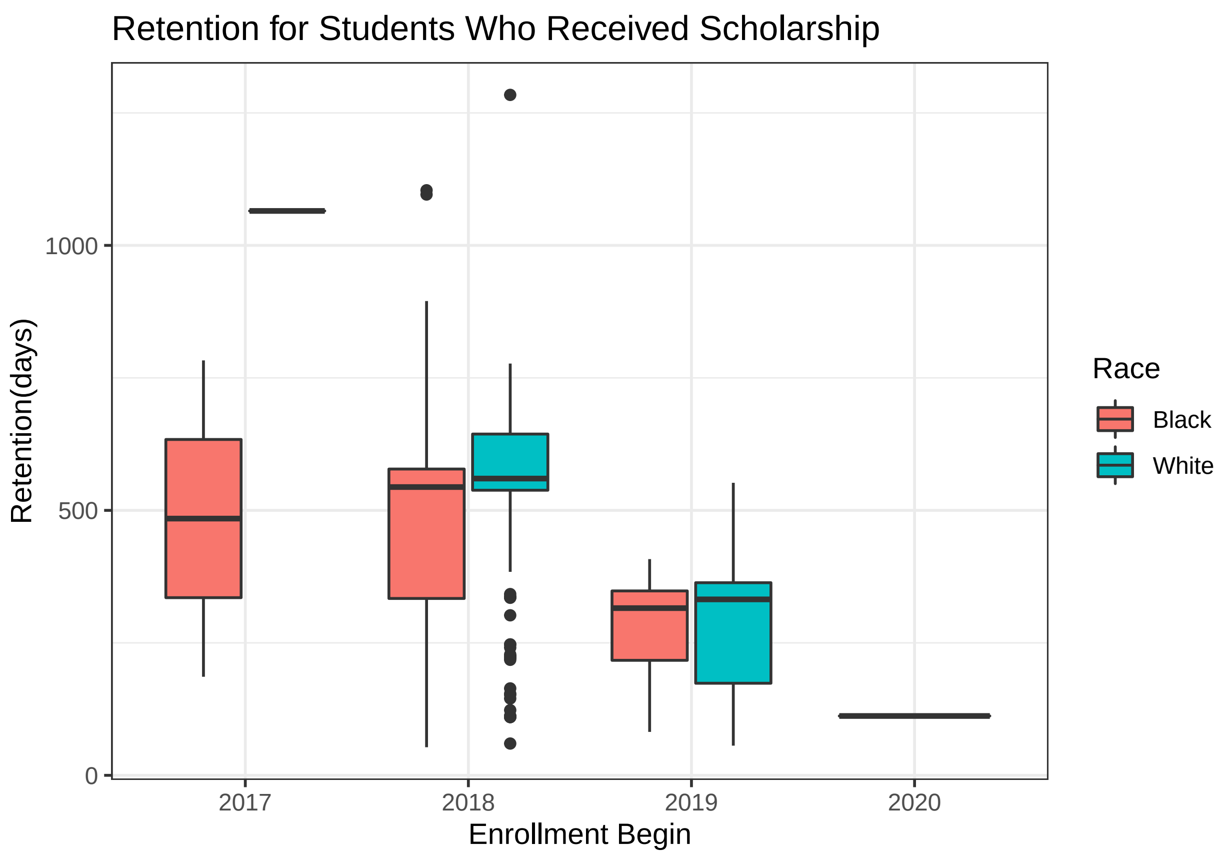
Findings:

- The differences are significant for both 2018 and 2019.
- Findings from significance tests align with our what we observe in the box plot above.
  - Well aligns with what we observe for 2018 enrollment year
  - Not perfectly aligns with what we see for 2019 enrollment year (i.e. no significant difference in retention between racial groups)

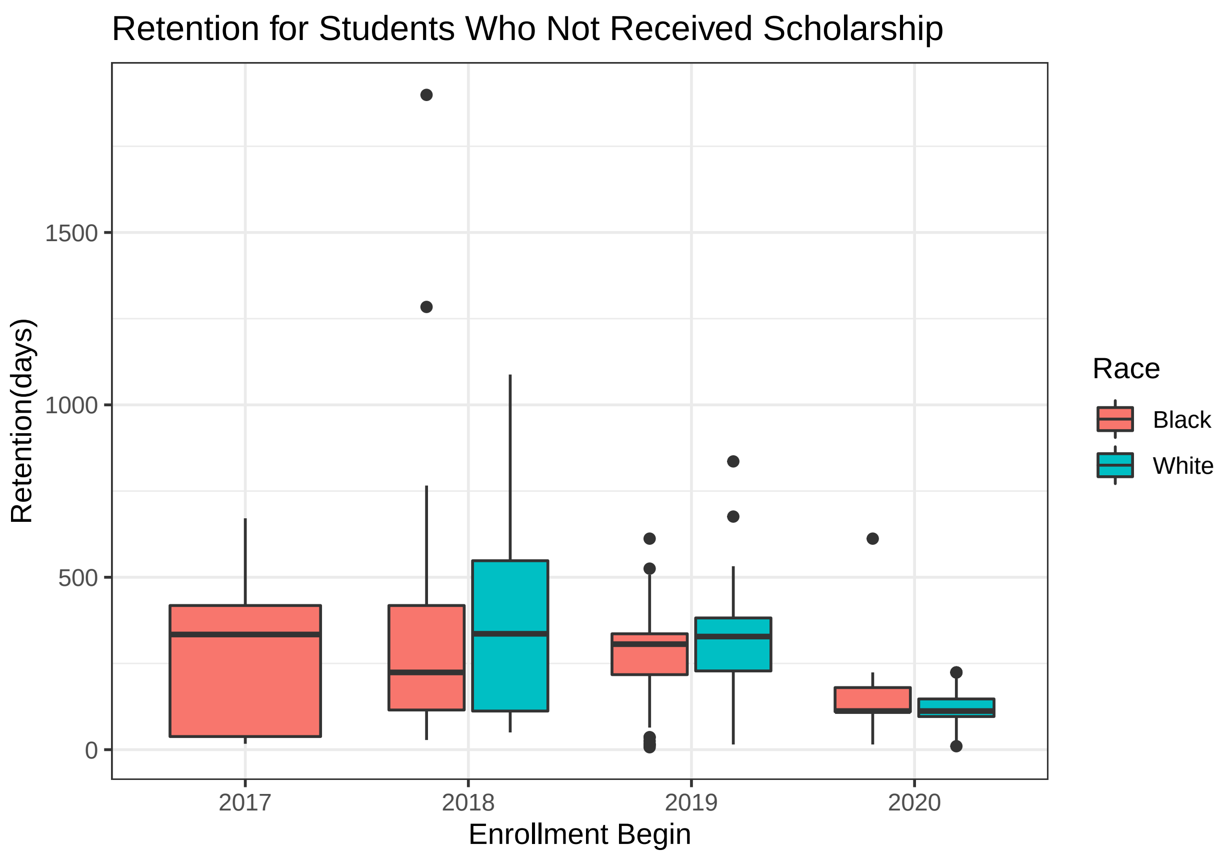
Comparison III: Retention vs. Interaction between Scholarship receipt and Race

EDA

```r
# Students who received the scholarship
# Visualization of retention comparison under race
nsc_demographics2 %>%
  filter(EverReceivedPromiseAward=="yes") %>%
  ggplot(aes(x=as.factor(ENROLLMENT_BEGIN_year),
             y=retention, fill=Race)) +
  geom_boxplot() +
  labs(x = "Enrollment Begin",
       y = "Retention(days)",
       title = "Retention for Students Who Received Scholarship") +
  theme_bw()
```




```r
# Filter students who did not receive the scholarship
# Visualization of retention comparison under race
nsc_demographics2 %>%
  filter(EverReceivedPromiseAward=="no") %>%
  ggplot(aes(x=as.factor(ENROLLMENT_BEGIN_year),
             y=retention,
             fill=Race)) +
  geom_boxplot() +
  labs(x = "Enrollment Begin",
       y = "Retention(days)",
       title = "Retention for Students Who Not Received Scholarship") +
  theme_bw()
```



Check for Significance for 2018 & 2019

```
# Prepare data sets for 2018 & 2019
nsc_demographics2_2018 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2018,
         EverReceivedPromiseAward == "yes")
nsc_demographics2_2019 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2019,
         EverReceivedPromiseAward == "yes")

# Use Bartlett test to check for equal variance or not (2018)
bartlett.test(retention~Race, data = nsc_demographics2_2018) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 3.2004, df = 1, p-value = 0.07362
```

```
# T-test to check for significance of difference in retention (2018)
oneway.test(retention~Race, data = nsc_demographics2_2018, var.equal = TRUE)
```

```
##
```

```
##  One-way analysis of means
##
## data:  retention and Race
## F = 13.073, num df = 1, denom df = 407, p-value = 0.0003371
```

```r
# Use Bartlett test to check for equal variance or not (2019)
bartlett.test(retention~Race, data = nsc_demographics2_2019) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 1.7403, df = 1, p-value = 0.1871
```

```r
# T-test to check for significance of difference in retention (2019)
oneway.test(retention~Race, data = nsc_demographics2_2019, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  retention and Race
## F = 0.44839, num df = 1, denom df = 38, p-value = 0.5071
```

```r
# Prepare data sets for 2018 & 2019
nsc_demographics3_2018 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2018,
         EverReceivedPromiseAward == "no")
nsc_demographics3_2019 = nsc_demographics2 %>%
  filter(ENROLLMENT_BEGIN_year == 2019,
         EverReceivedPromiseAward == "no")

# Use Bartlett test to check for equal variance or not (2018)
bartlett.test(retention~Race, data = nsc_demographics3_2018) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 0.79106, df = 1, p-value = 0.3738
```

```r
# T-test to check for significance of difference in retention (2018)
oneway.test(retention~Race, data = nsc_demographics3_2018, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  retention and Race
## F = 0.38341, num df = 1, denom df = 95, p-value = 0.5373
```

```r
# Use Bartlett test to check for equal variance or not (2019)
bartlett.test(retention~Race, data = nsc_demographics3_2019) #equal variance
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  retention by Race
## Bartlett's K-squared = 0.018019, df = 1, p-value = 0.8932
```

```r
# T-test to check for significance of difference in retention (2019)
oneway.test(retention~Race, data = nsc_demographics3_2019, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  retention and Race
## F = 29.444, num df = 1, denom df = 551, p-value = 8.633e-08
```

Findings:

- Among students who received the scholarship, we see that for 2018, there is a significant racial difference, but for 2019 the difference is not significant.
- Among students who did not receive the scholarship, we see reverse happens, the difference is not significant for 2018 but significant for 2019.

Part II: Multivariate Regression Analysis for Whole Range

Data Wrangling

```
# Import data sets
# data_qualified is the final data set used in logistic regression modeling from scholarship analysis
# retention_demographics is the joined data set between NSC and demographics
data_qualified <- read_csv("data_qualified.csv")
retention_demographics <- read_csv("retention_demographics.csv") %>% dplyr::select(-c(1))
retention_demographics$EverReceivedPromiseAward <-
  ifelse(retention_demographics$EverReceivedPromiseAward == "yes", 1, 0)
```

```
# Inner join data_qualified and retention_demographics to obtain data set used for regression
retention_reg <- inner_join(data_qualified, retention_demographics, by = c("RandomID",
                                                                           "Race",
                                                                           "GradYear",
                                                                           "EverReceivedPromiseAward"))

# Categorize race as white, black, and other
retention_reg$Race <- ifelse(retention_reg$Race != "White" & retention_reg$Race != "African American",
                             "Other",
                             retention_reg$Race)
```

First Trial on Linear Regression

Linear Regression for Students Starting College in 2018

```
# Filter students who started college in 2018
retention2018 <- retention_reg %>%
  filter(ENROLLMENT_BEGIN_year == 2018) #485 obs
```

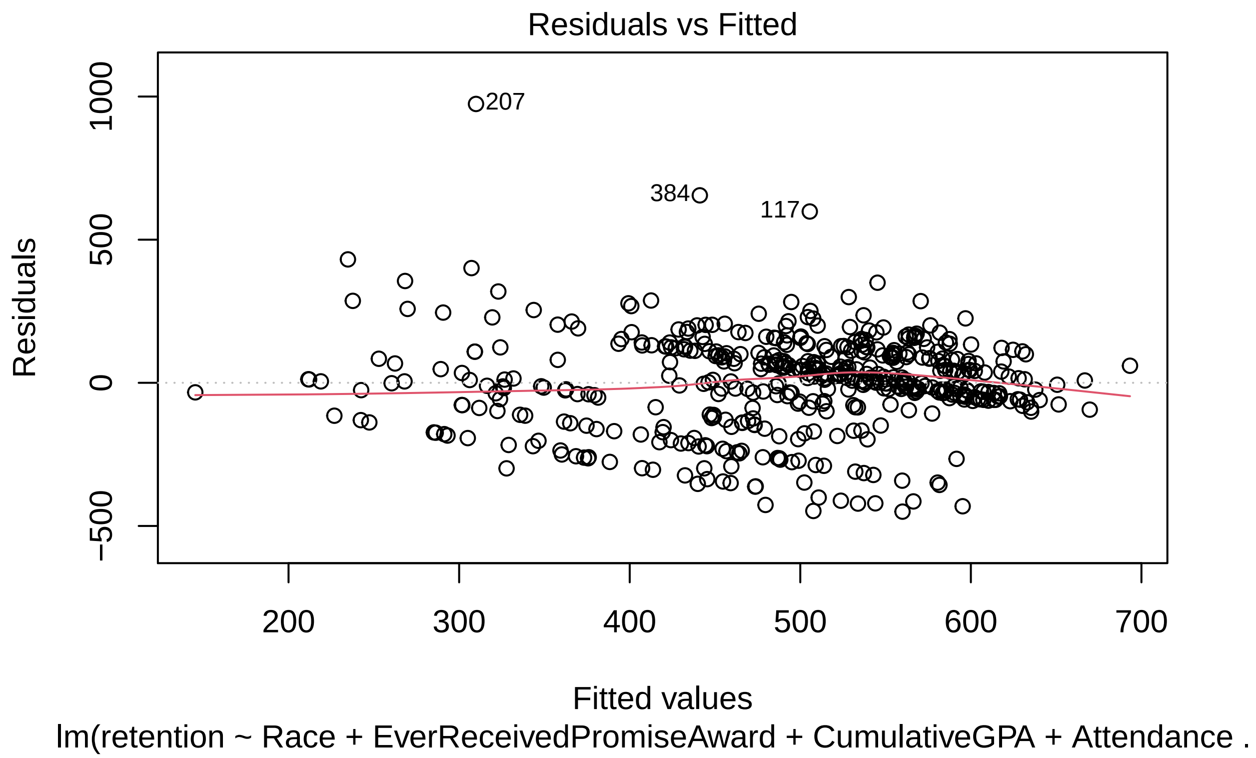
```
# Construct null and full model for linear regression 2018
md.null2018 = lm(retention ~ Race + EverReceivedPromiseAward + Race*EverReceivedPromiseAward,
                 data = retention2018)
md.full2018 = lm(retention~AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race+Gender+ELLStatus+
                   IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd+
                   EverReceivedPromiseAward+QualifiedforCorePromise+semester+
                   Race*EverReceivedPromiseAward, data = retention2018)
```

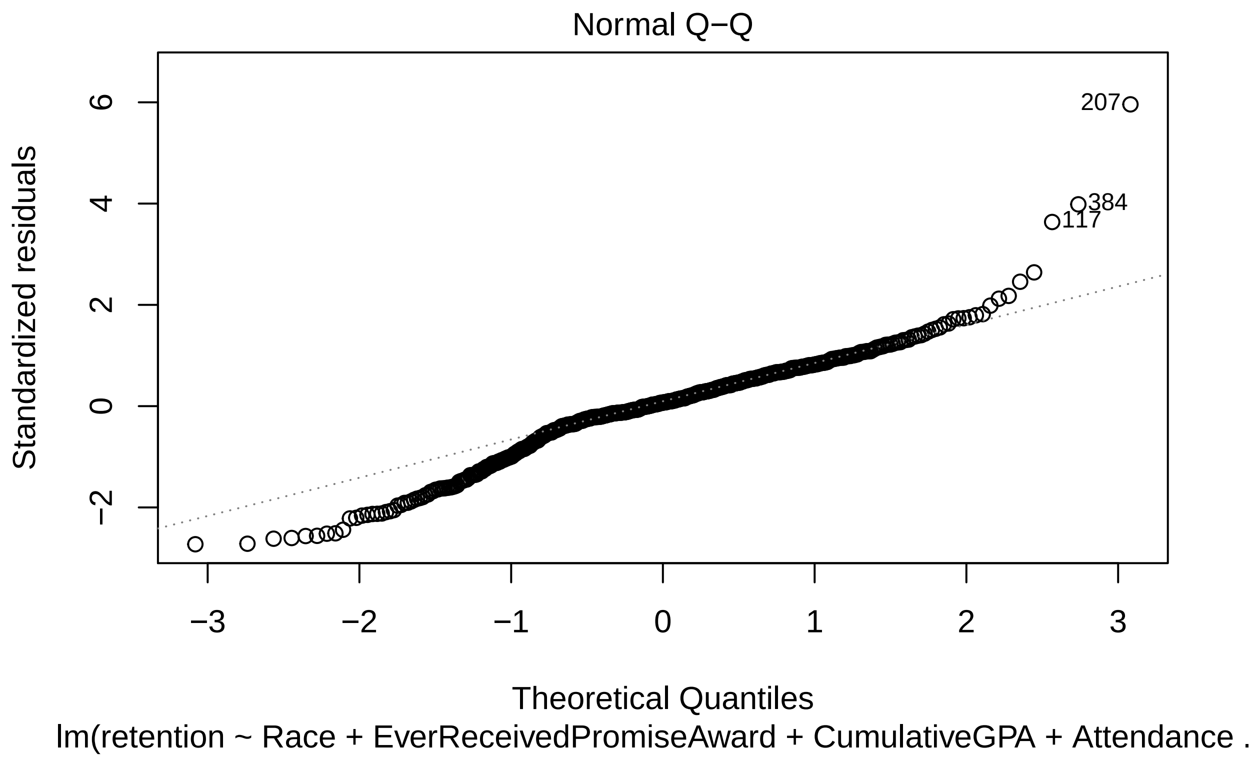
```
# Stepwise variable selection (both directions)
md.2018 = step(md.null2018,
               list(lower = formula(md.null2018),
                    upper = formula(md.full2018)),
               direction = "both", trace = 0)
summary(md.2018)
```

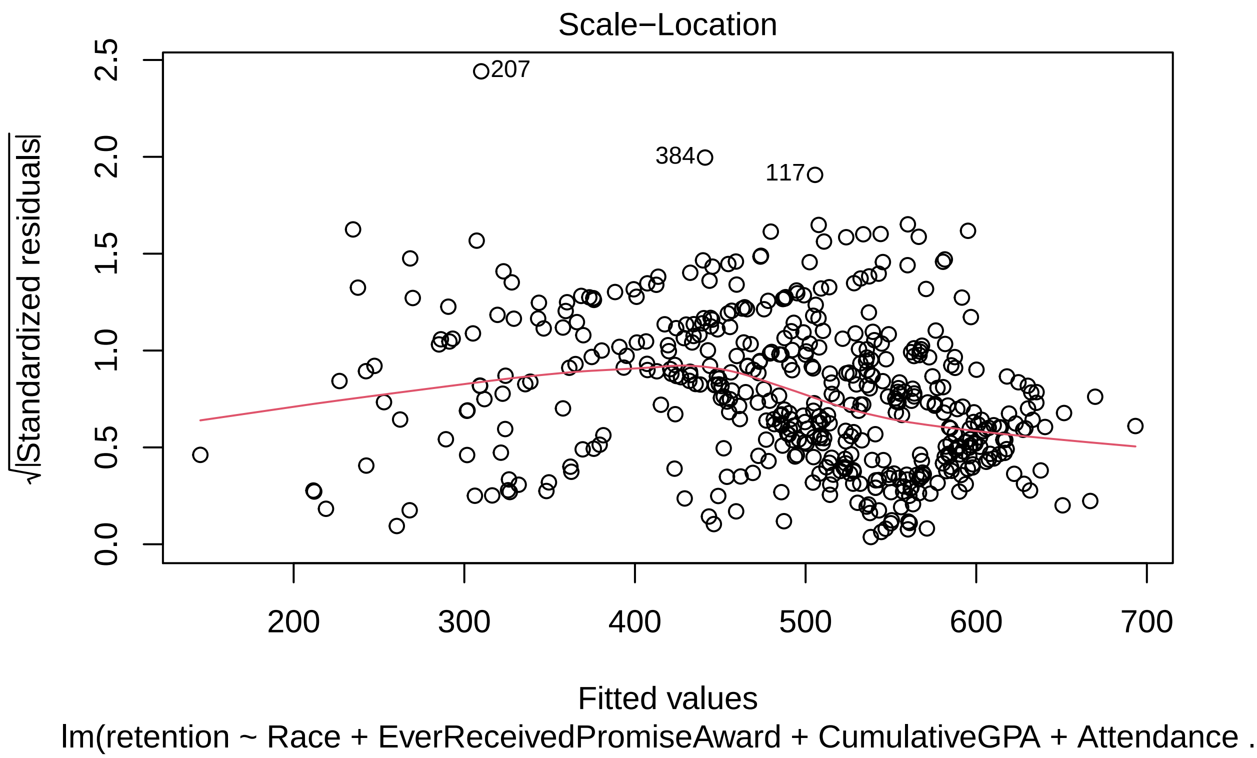
```
##
## Call:
## lm(formula = retention ~ Race + EverReceivedPromiseAward + CumulativeGPA +
```

```
##     AttendanceRate + Gender + SAT_Total + KeystoneMean + Num_CTE +
##     Race:EverReceivedPromiseAward, data = retention2018)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -449.94  -68.04   11.33   99.33  974.10
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          568.52533  491.96068   1.156 0.248415
## RaceOther                             22.48731   58.39909   0.385 0.700364
## RaceWhite                              6.72192   46.56124   0.144 0.885272
## EverReceivedPromiseAward             117.31801   28.94333   4.053  5.9e-05 ***
## CumulativeGPA                         71.52453   22.64271   3.159 0.001685 **
## AttendanceRate                       878.09592  277.32582   3.166 0.001644 **
## GenderMale                           -34.40616   16.54698  -2.079 0.038128 *
## SAT_Total                              0.27647    0.08116   3.407 0.000714 ***
## KeystoneMean                          -1.01836    0.33855  -3.008 0.002770 **
## Num_CTE                              -14.34949    9.39897  -1.527 0.127502
## RaceOther:EverReceivedPromiseAward    -2.34132   64.03626  -0.037 0.970849
## RaceWhite:EverReceivedPromiseAward     2.41024   48.48180   0.050 0.960371
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 165.7 on 473 degrees of freedom
## Multiple R-squared:  0.2545, Adjusted R-squared:  0.2371
## F-statistic: 14.68 on 11 and 473 DF,  p-value: < 2.2e-16
```

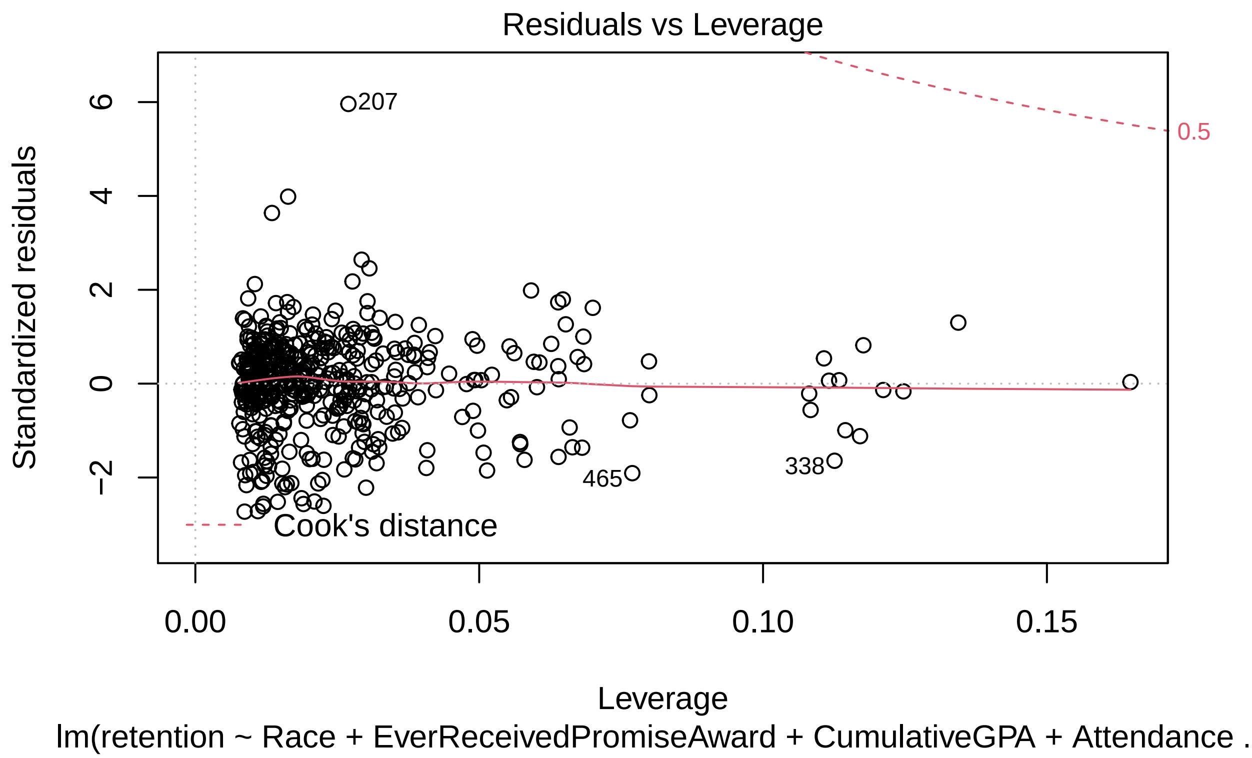
```r
# Model diagnostics
plot(md.2018)
```


Residuals vs Fitted
Residuals
Fitted values
lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + Attendance .
207
384
117
1000
500
0
−500
200
300
400
500
600
700


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + Attendance .
207
384
117
−3
−2
−1
0
1
2
3
6
4
2
0
−2



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
207
384
117
200
300
400
500
600
700
Fitted values
lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + Attendance .



Linear Regression for Students Starting College in 2019

```
retention2019 <- retention_reg %>%
  filter(ENROLLMENT_BEGIN_year == 2019) #663 obs
```

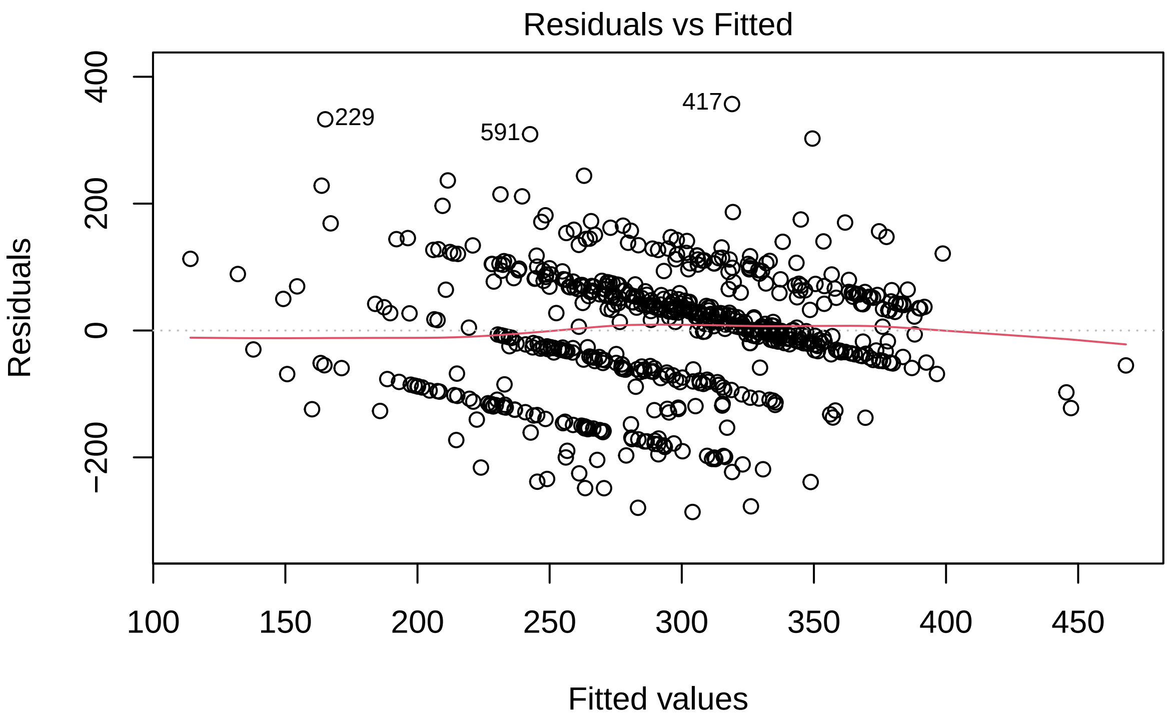
```
# Construct null and full model for linear regression 2019
md.null2019 = lm(retention ~ Race + EverReceivedPromiseAward + Race*EverReceivedPromiseAward,
                 data = retention2019)
md.full2019 = lm(retention~AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race+Gender+ELLStatus+
                   IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd+
                   EverReceivedPromiseAward+QualifiedforCorePromise+semester+
                   Race*EverReceivedPromiseAward, data = retention2019)
```

```
# Stepwise variable selection (both directions)
md.2019 = step(md.null2019,
               list(lower = formula(md.null2019),
                    upper = formula(md.full2019)),
               direction = "both", trace = 0)
summary(md.2019)
```

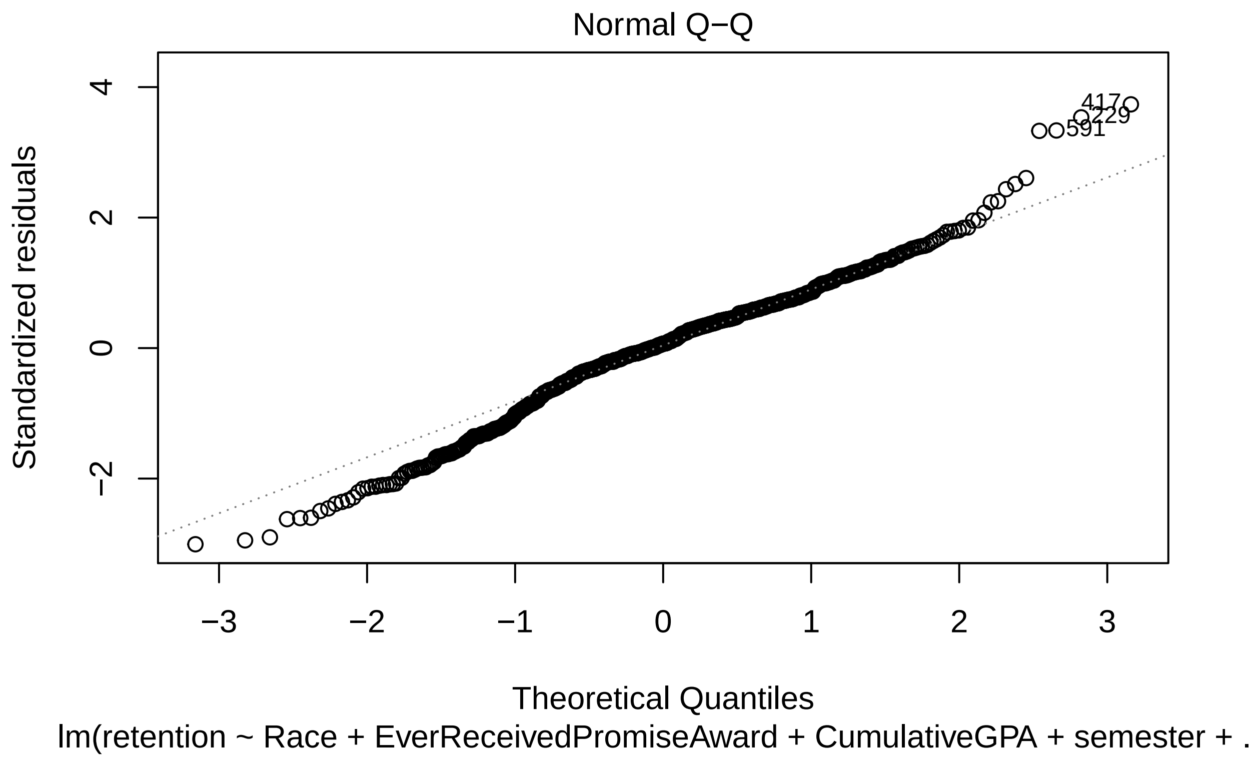
```
##
## Call:
## lm(formula = retention ~ Race + EverReceivedPromiseAward + CumulativeGPA +
```

```
##     semester + SAT_Total + Gender + ELLStatus + Race:EverReceivedPromiseAward,
##     data = retention2019)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -286.08  -51.18    6.52   59.19  356.98
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          -7.03681   33.11796  -0.212  0.83180
## RaceOther                           -12.65312   13.11819  -0.965  0.33515
## RaceWhite                            -0.16154    9.94309  -0.016  0.98704
## EverReceivedPromiseAward             11.22200   24.74243   0.454  0.65031
## CumulativeGPA                        59.32630    9.13924   6.491 1.74e-10 ***
## semesterSpring                      -66.25741   16.80158  -3.944 8.94e-05 ***
## SAT_Total                             0.08155    0.02621   3.111  0.00195 **
## GenderMale                          -14.80311    8.04102  -1.841  0.06611 .
## ELLStatusNot in ELL                  44.55902   25.89806   1.721  0.08583 .
## RaceOther:EverReceivedPromiseAward  98.87681   38.34957   2.578  0.01016 *
## RaceWhite:EverReceivedPromiseAward -23.19223   33.06471  -0.701  0.48330
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 95.82 on 622 degrees of freedom
## Multiple R-squared:  0.2289, Adjusted R-squared:  0.2165
## F-statistic: 18.46 on 10 and 622 DF,  p-value: < 2.2e-16
```

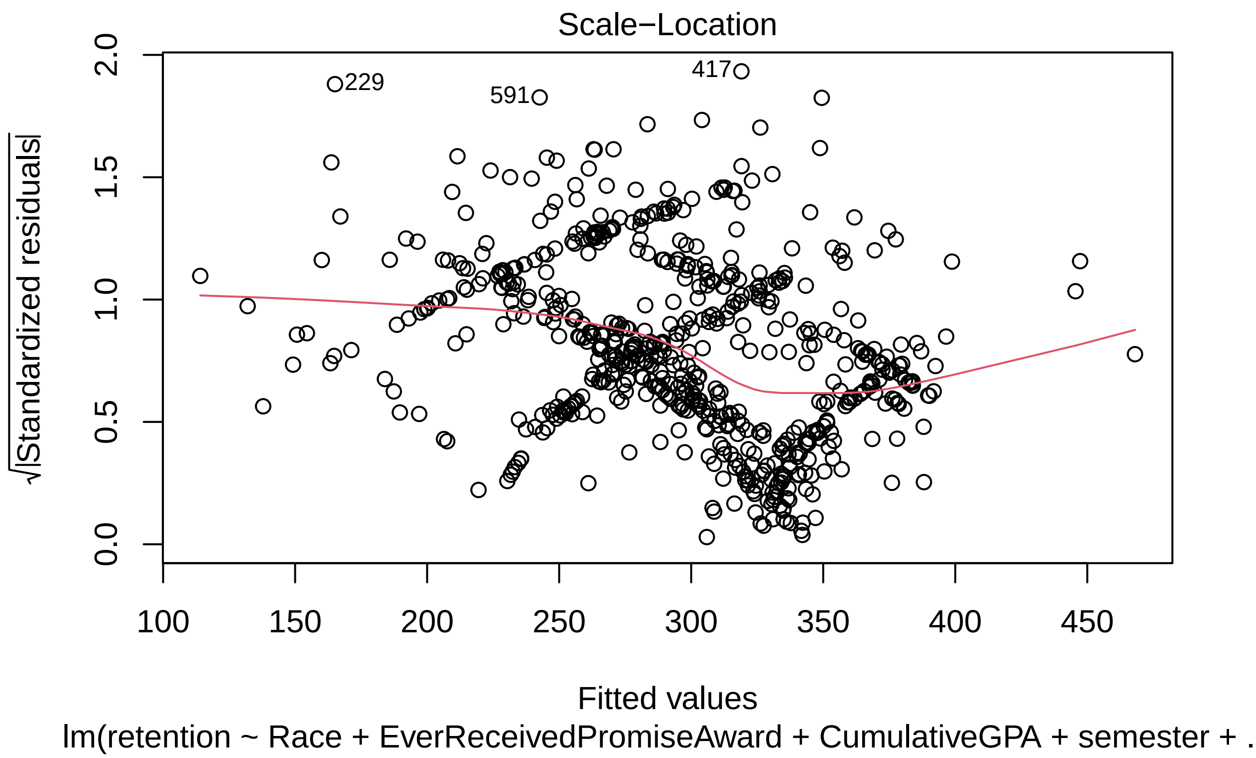
```r
# Model diagnostics
plot(md.2019)
```

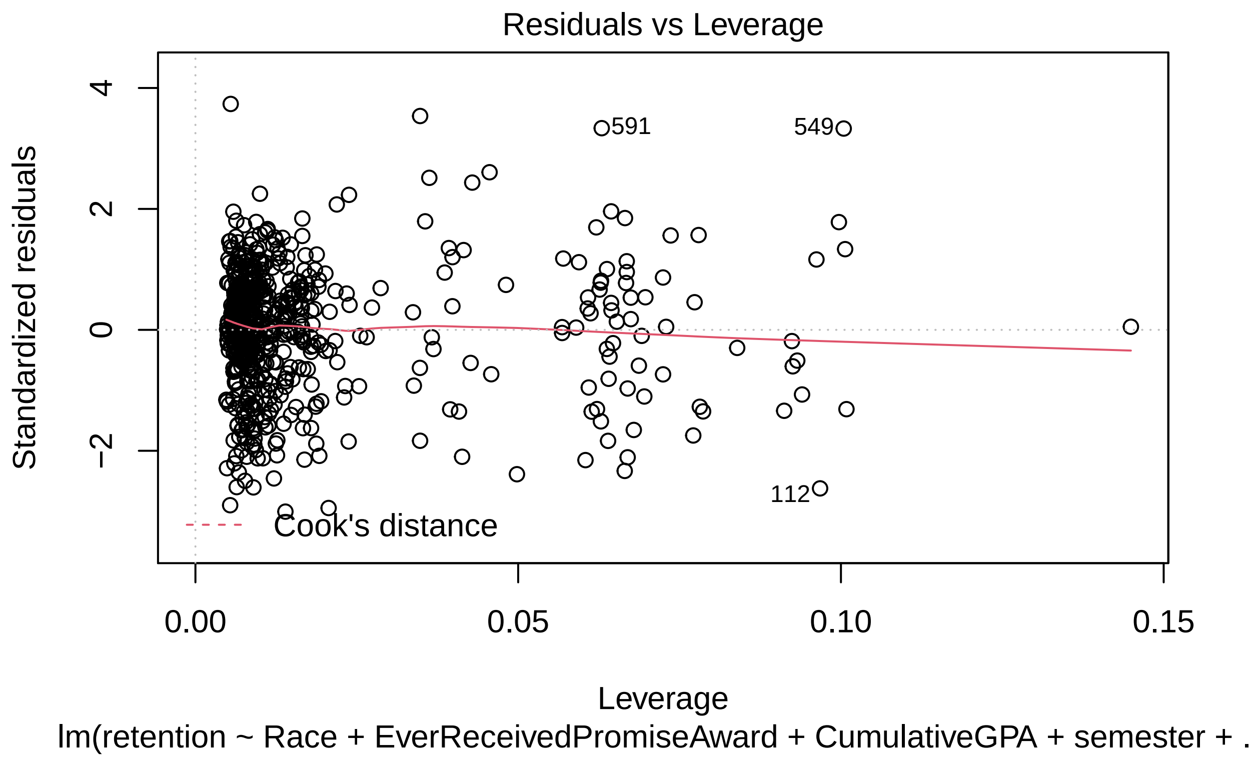


lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + semester + .



Normal Q−Q
Standardized residuals
4
2
0
−2
417
229
591
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + semester + .


Scale−Location
√|Standardized residuals|
Fitted values
lm(retention ~ Race + EverReceivedPromiseAward + CumulativeGPA + semester + .
229
591
417
0.0
0.5
1.0
1.5
2.0
100
150
200
250
300
350
400
450



Second Trial on Poisson Regression

Poisson Regression for Students Starting College in 2018

```
# Construct null possion model and full poisson model
poisson2018.null = glm(retention~Race+EverReceivedPromiseAward+Race*EverReceivedPromiseAward,
                       family = "poisson", data = retention2018)
poisson2018.full = glm(retention~AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race+Gender+
                         ELLStatus+IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd+
                         EverReceivedPromiseAward+QualifiedforCorePromise+semester+
                         Race*EverReceivedPromiseAward, family = "poisson", data = retention2018)
```

```
# Stepwise variable selection (both directions)
poisson2018 = step(poisson2018.full,
                   list(lower = formula(poisson2018.null),
                        upper = formula(poisson2018.full)),
                   direction = "both",
                   trace = 0)
summary(poisson2018)
```

```
##
## Call:
## glm(formula = retention ~ AttendanceRate + Num_AP + Num_CTE +
```

```
##     KeystoneMean + Race + Gender + ELLStatus + IEPGroup + EconDisad +
##     SAT_Total + CumulativeGPA + MagnetInd + EverReceivedPromiseAward +
##     QualifiedforCorePromise + semester + Race * EverReceivedPromiseAward,
##     family = "poisson", data = retention2018)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -23.933   -3.585    0.410    4.477   38.736
##
## Coefficients:
##                                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          6.110e+00  1.441e-01  42.393  < 2e-16 ***
## AttendanceRate                       2.246e+00  8.707e-02  25.800  < 2e-16 ***
## Num_AP                               4.585e-03  1.041e-03   4.403 1.07e-05 ***
## Num_CTE                             -2.825e-02  2.743e-03 -10.300  < 2e-16 ***
## KeystoneMean                        -2.166e-03  9.581e-05 -22.603  < 2e-16 ***
## RaceOther                            4.463e-02  2.040e-02   2.187  0.02873 *
## RaceWhite                            6.300e-02  1.500e-02   4.199 2.68e-05 ***
## GenderMale                          -7.107e-02  4.611e-03 -15.415  < 2e-16 ***
## ELLStatusNot in ELL                 -5.165e-02  1.620e-02  -3.189  0.00143 **
## IEPGroupIEP                          2.181e-02  1.123e-02   1.942  0.05217 .
## IEPGroupNot IEP or Gifted           -5.117e-02  6.443e-03  -7.942 1.99e-15 ***
## EconDisadRegular Lunch               1.093e-02  4.630e-03   2.361  0.01822 *
## SAT_Total                            4.265e-04  2.367e-05  18.015  < 2e-16 ***
## CumulativeGPA                        1.697e-01  7.631e-03  22.244  < 2e-16 ***
## MagnetInd                            3.736e-02  4.682e-03   7.979 1.47e-15 ***
## EverReceivedPromiseAward             3.162e-01  9.688e-03  32.636  < 2e-16 ***
## QualifiedforCorePromiseyes          -4.058e-02  9.452e-03  -4.293 1.76e-05 ***
## semesterSpring                       1.279e-01  4.022e-02   3.180  0.00147 **
## RaceOther:EverReceivedPromiseAward -4.254e-03  2.116e-02  -0.201  0.84068
## RaceWhite:EverReceivedPromiseAward -4.717e-02  1.540e-02  -3.064  0.00219 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 41127  on 484  degrees of freedom
## Residual deviance: 31222  on 465  degrees of freedom
## AIC: 35106
##
## Number of Fisher Scoring iterations: 5
```

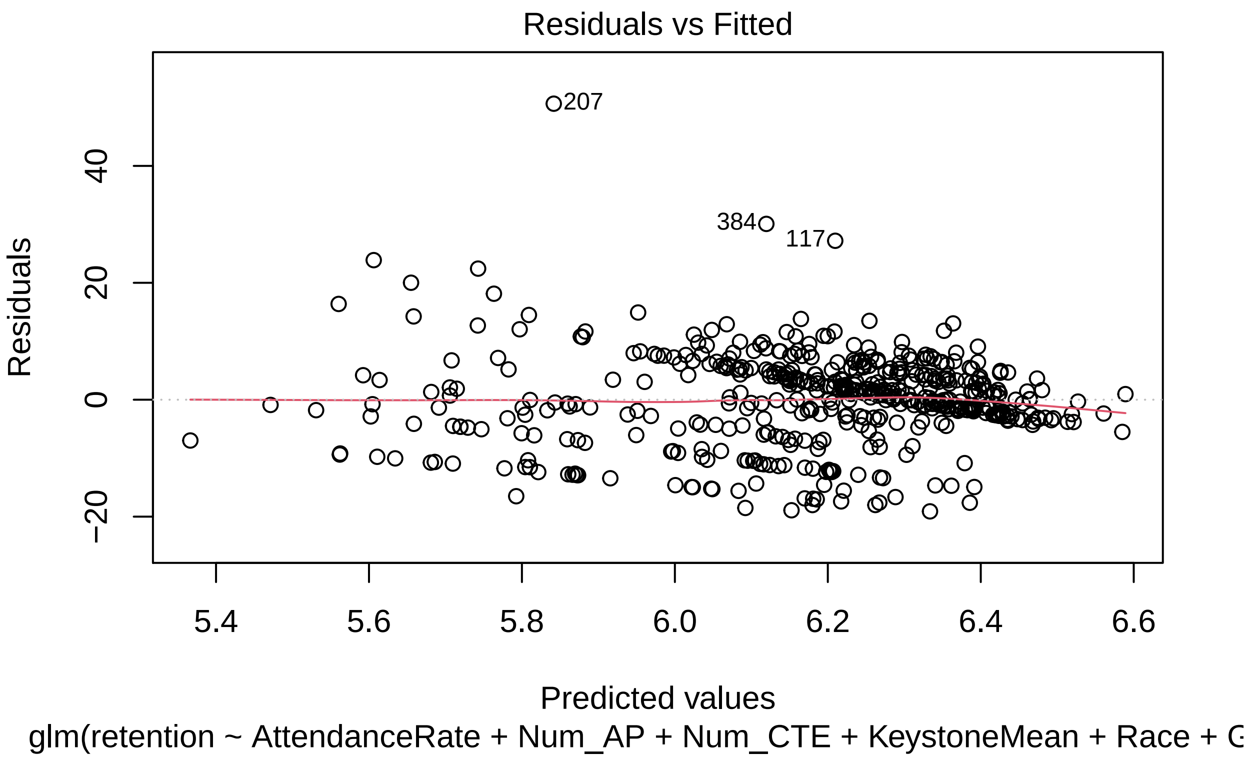
```r
# Model diagnostics
# Residual deviance
pchisq(poisson2018$deviance,
       poisson2018$df.residual,
       lower.tail = FALSE)
```

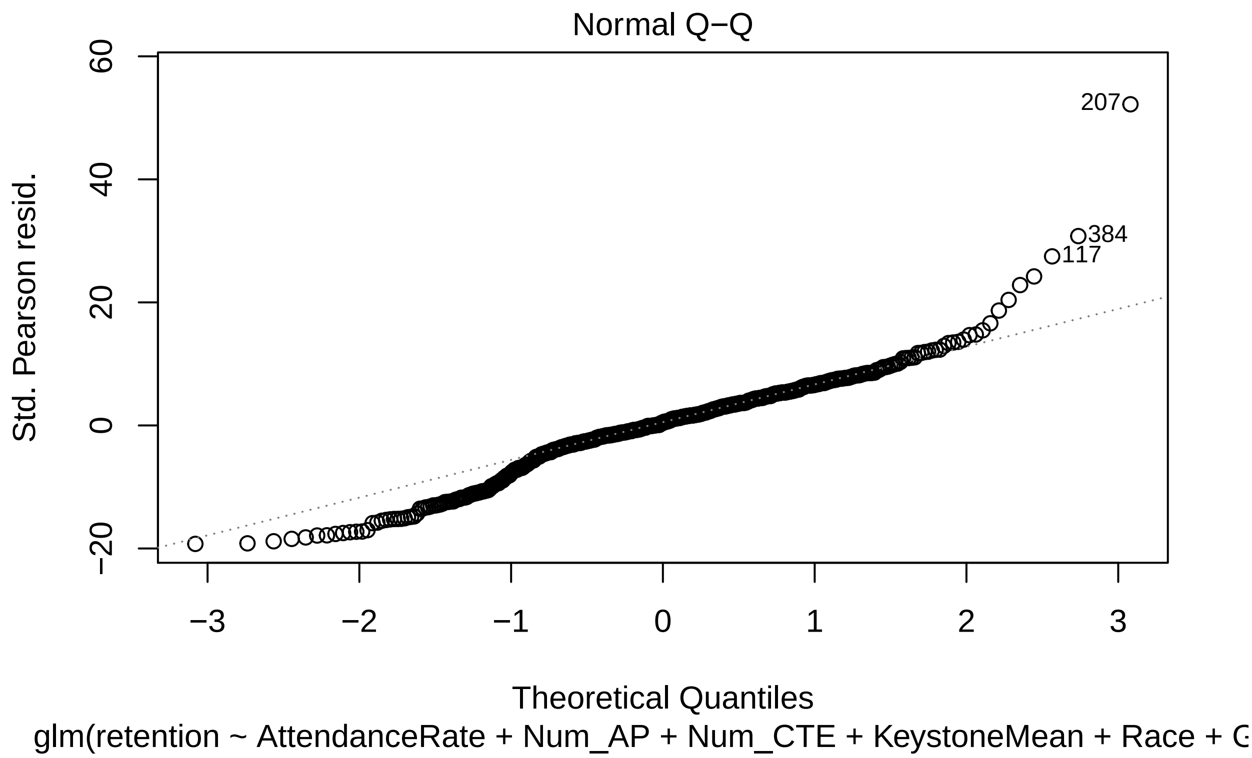
```
## [1] 0
```

```r
# Diagnostics plot
plot(poisson2018)
```

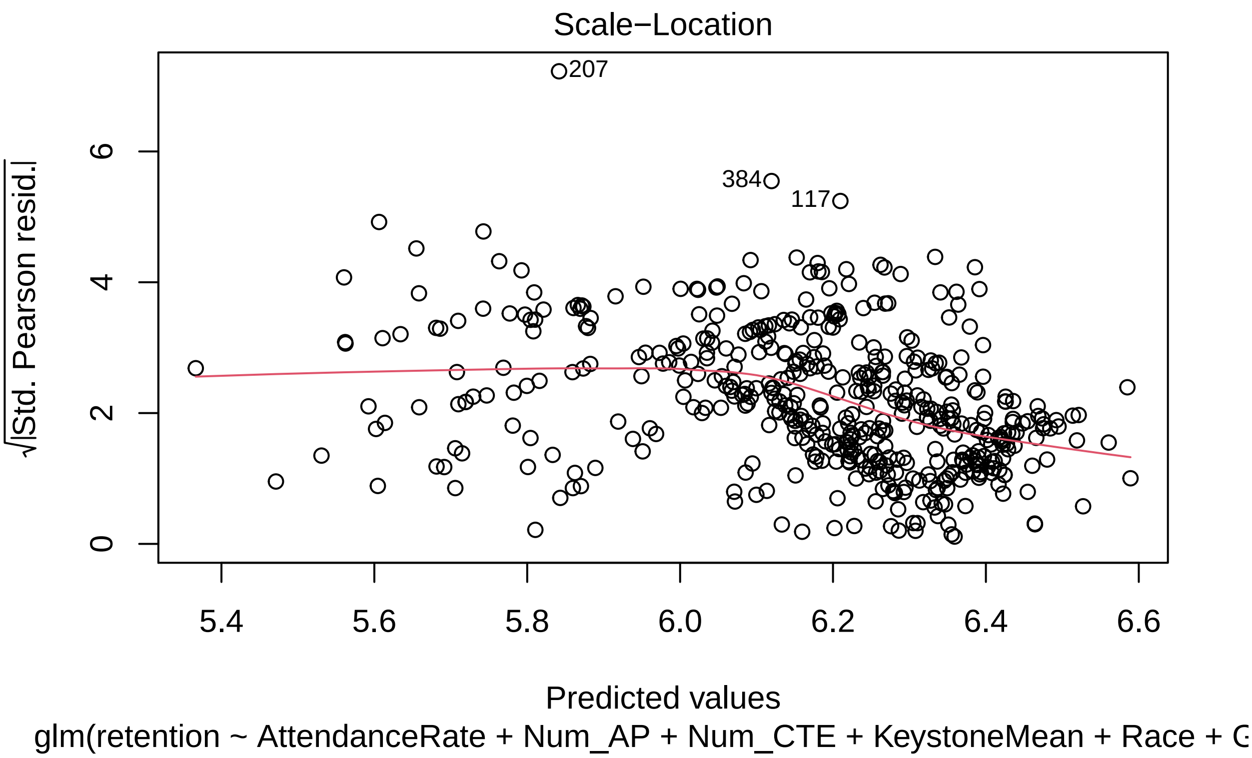
```
## Warning: not plotting observations with leverage one:
```

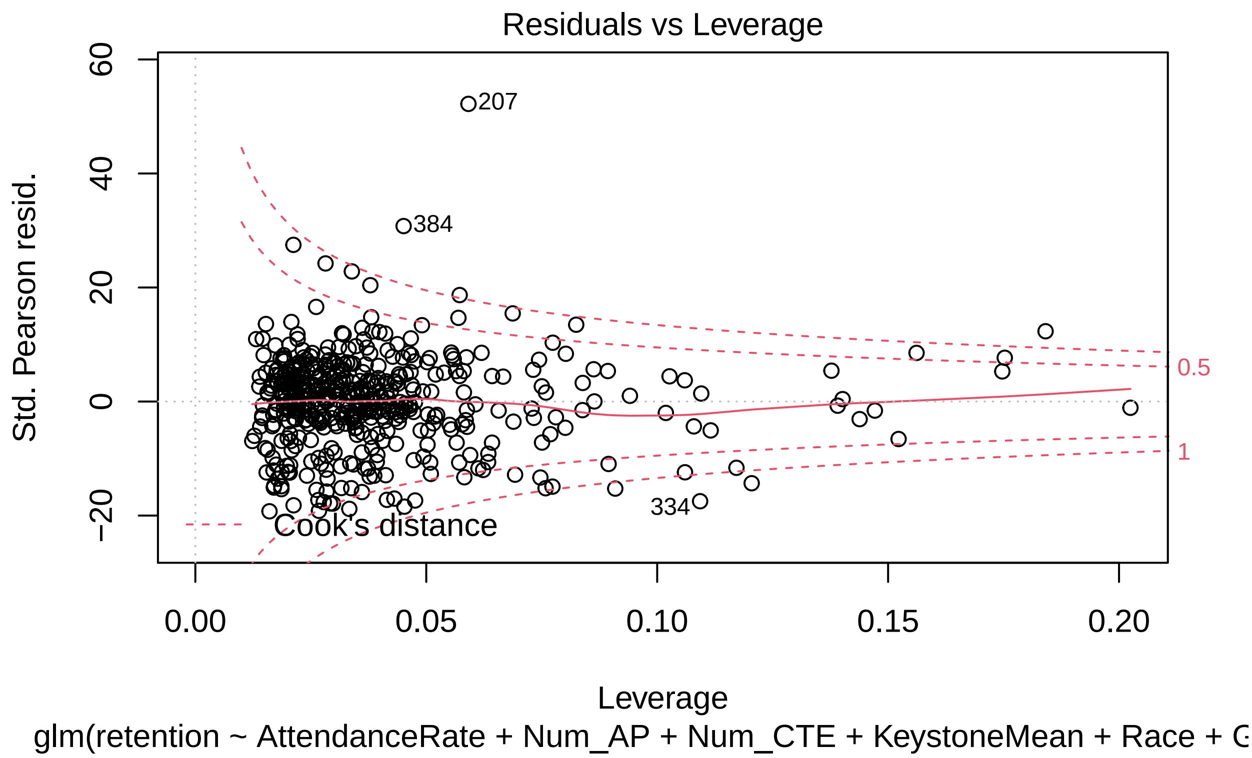
```
## 171
```


Residuals vs Fitted
207
384
117
Residuals
40
20
0
−20
5.4
5.6
5.8
6.0
6.2
6.4
6.6
Predicted values
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G

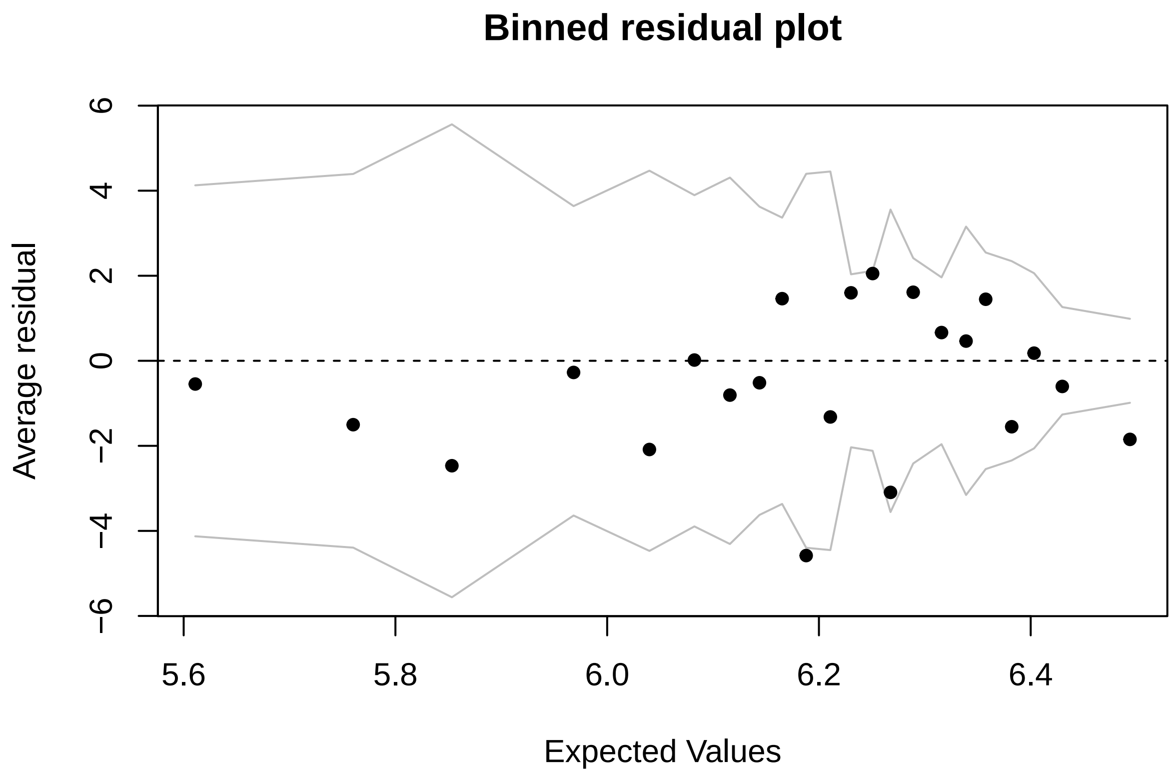


Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G
207
384
117
60
40
20
0
−20
−3
−2
−1
0
1
2
3


Scale−Location
207
384
117
√|Std. Pearson resid.|
0
2
4
6
5.4
5.6
5.8
6.0
6.2
6.4
6.6
Predicted values
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G




```r
# Binned plot
x.2018 = predict(poisson2018)
y.2018 = residuals(poisson2018)
binnedplot(x.2018,y.2018)
```



Poisson Regression for Students Starting College in 2019

```
# Construct null possion model and full poisson model
poisson2019.null = glm(retention~Race+EverReceivedPromiseAward+Race*EverReceivedPromiseAward,
                       family = "poisson", data = retention2019)
poisson2019.full = glm(retention~AttendanceRate+Num_AP+Num_CTE+KeystoneMean+Race+Gender+
                         ELLStatus+IEPGroup+EconDisad+SAT_Total+CumulativeGPA+MagnetInd+
                         EverReceivedPromiseAward+QualifiedforCorePromise+semester+
                         Race*EverReceivedPromiseAward, family = "poisson", data = retention2019)
```

```
# Stepwise variable selection (both directions)
poisson2019 = step(poisson2019.full, list(lower = formula(poisson2019.null),
                                          upper = formula(poisson2019.full)),
                   direction = "both", trace = 0)
summary(poisson2019)
```

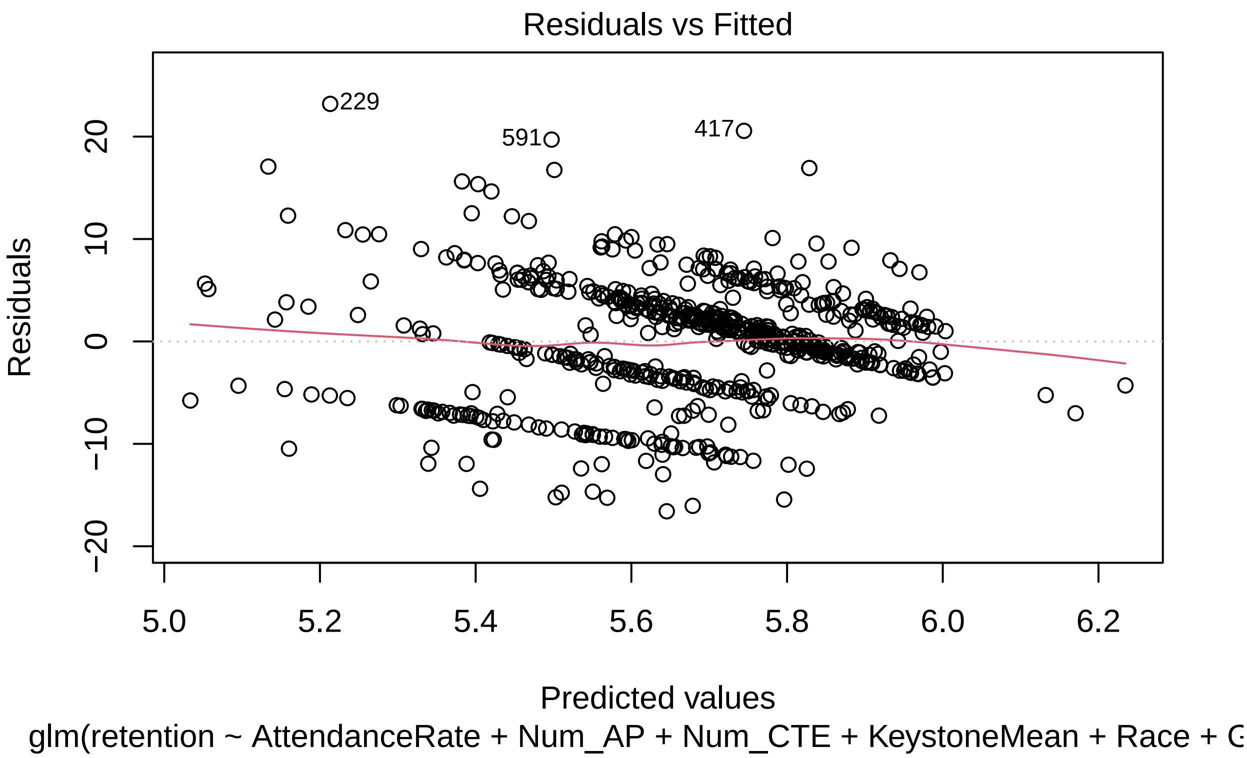
```
##
## Call:
## glm(formula = retention ~ AttendanceRate + Num_AP + Num_CTE +
##     KeystoneMean + Race + Gender + ELLStatus + EconDisad + SAT_Total +
##     CumulativeGPA + MagnetInd + EverReceivedPromiseAward + QualifiedforCorePromise +
##     semester + Race:EverReceivedPromiseAward, family = "poisson",
##     data = retention2019)
```

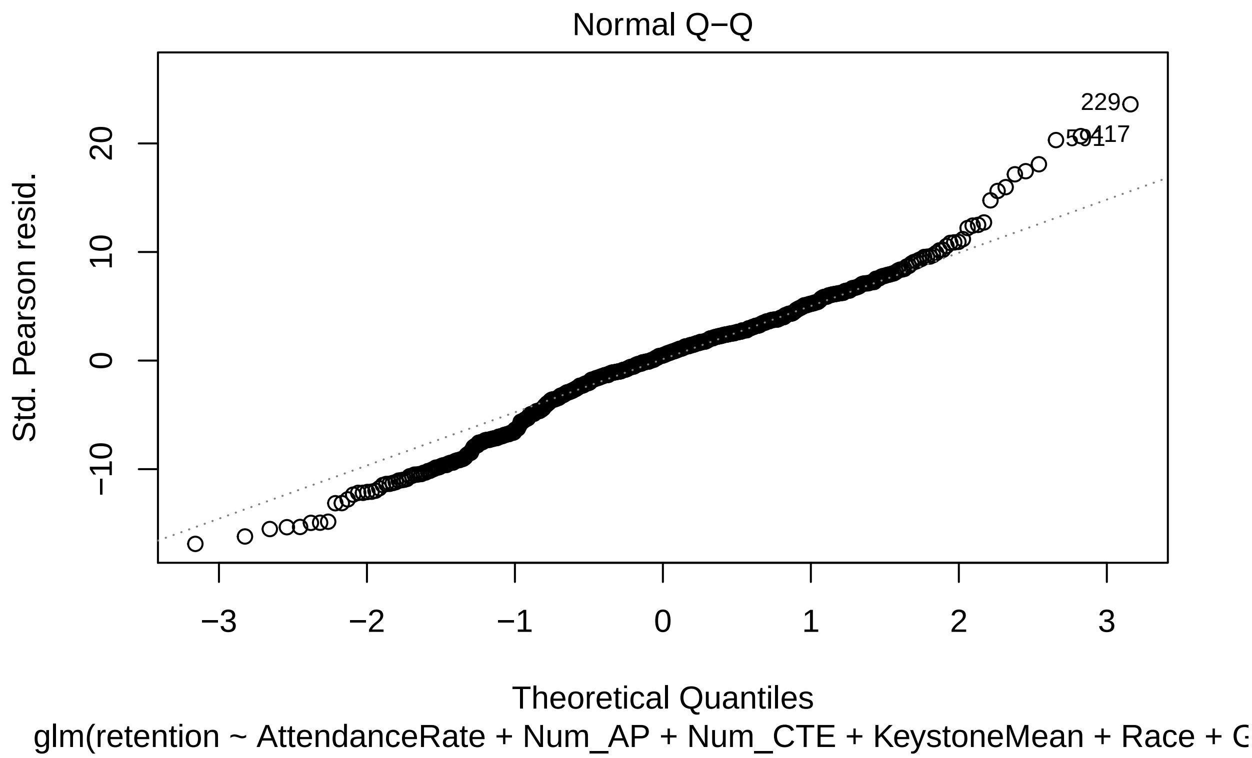
```
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -22.8894   -3.1941    0.4624    3.2503   19.0998
##
## Coefficients:
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                         4.432e+00  1.604e-01  27.634  < 2e-16 ***
## AttendanceRate                      6.489e-01  8.490e-02   7.644 2.11e-14 ***
## Num_AP                              3.395e-03  1.218e-03   2.787  0.00531 **
## Num_CTE                            -1.305e-02  2.532e-03  -5.152 2.57e-07 ***
## KeystoneMean                       -2.601e-04  1.077e-04  -2.415  0.01575 *
## RaceOther                          -3.699e-02  8.317e-03  -4.447 8.71e-06 ***
## RaceWhite                           9.875e-04  6.405e-03   0.154  0.87748
## GenderMale                         -5.275e-02  5.012e-03 -10.524  < 2e-16 ***
## ELLStatusNot in ELL                 1.950e-01  1.834e-02  10.636  < 2e-16 ***
## EconDisadRegular Lunch              2.151e-02  5.154e-03   4.174 3.00e-05 ***
## SAT_Total                           2.291e-04  2.543e-05   9.011  < 2e-16 ***
## CumulativeGPA                       1.701e-01  7.749e-03  21.951  < 2e-16 ***
## MagnetInd                           2.889e-02  5.126e-03   5.636 1.74e-08 ***
## EverReceivedPromiseAward            2.282e-02  1.587e-02   1.438  0.15039
## QualifiedforCorePromiseyes          6.454e-02  9.687e-03   6.663 2.69e-11 ***
## semesterSpring                     -2.563e-01  1.147e-02 -22.343  < 2e-16 ***
## RaceOther:EverReceivedPromiseAward  3.113e-01  2.271e-02  13.703  < 2e-16 ***
## RaceWhite:EverReceivedPromiseAward -5.807e-02  2.075e-02  -2.798  0.00514 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 29317  on 632  degrees of freedom
## Residual deviance: 23250  on 615  degrees of freedom
## AIC: 27983
##
## Number of Fisher Scoring iterations: 5
```

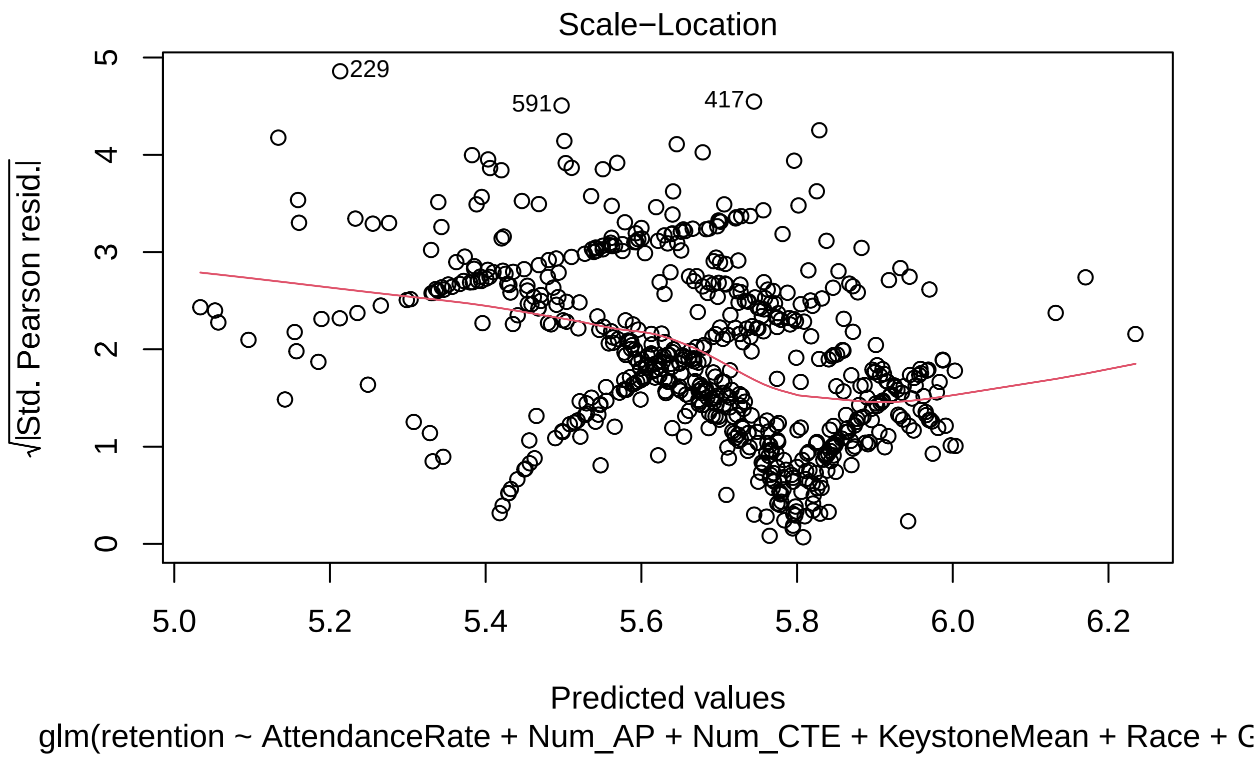
```r
# Model diagnostics
# Residual deviance
pchisq(poisson2019$deviance, poisson2019$df.residual, lower.tail = FALSE)
```

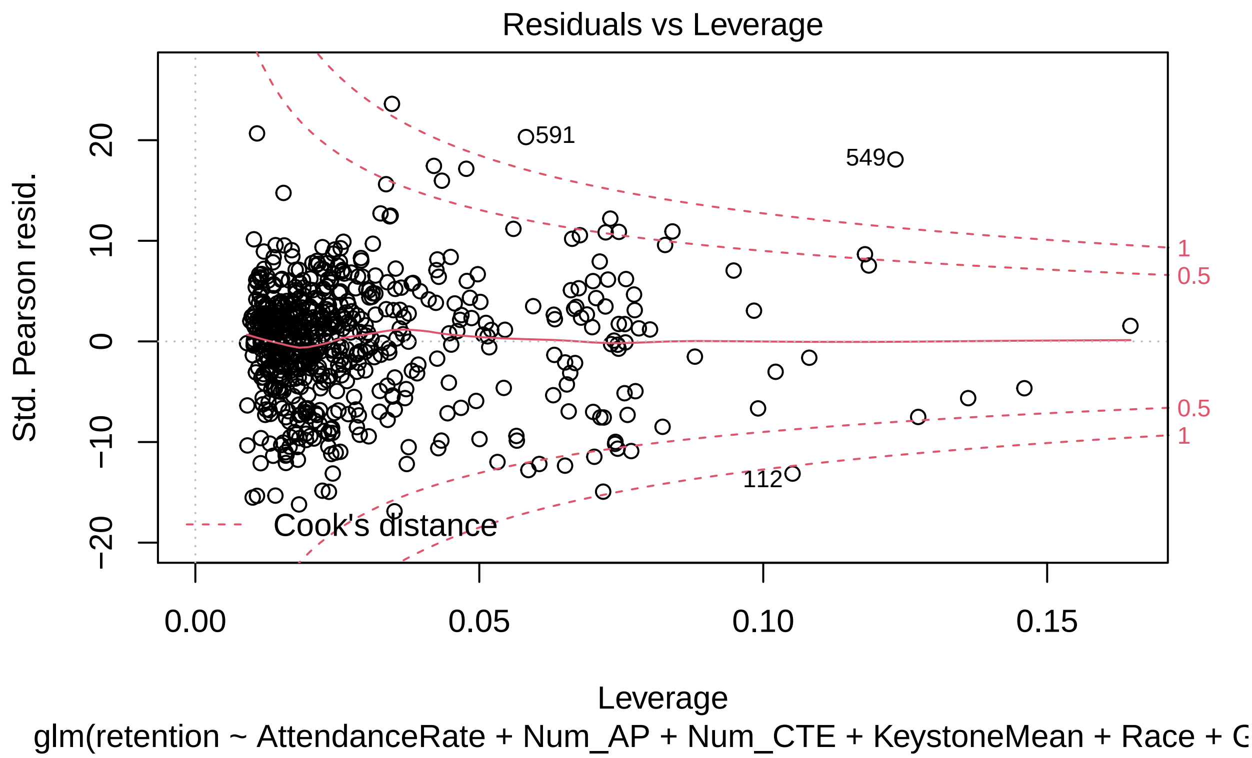
```
## [1] 0
```

```r
# Diagnostics plot
plot(poisson2019)
```

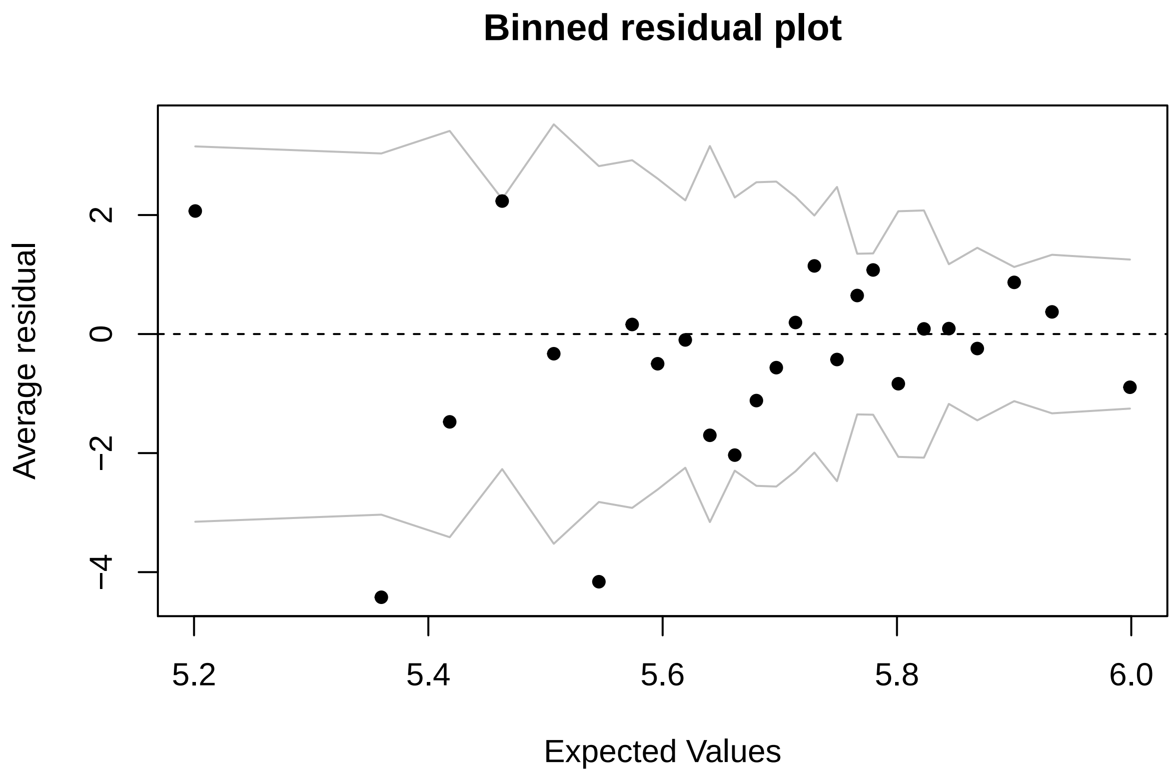

Residuals vs Fitted
229
591
417
Residuals
Predicted values
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G


Normal Q−Q
229
59417
591
Std. Pearson resid.
20
10
0
−10
−3
−2
−1
0
1
2
3
Theoretical Quantiles
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G


Scale−Location
229
591
417
√|Std. Pearson resid.|
0
1
2
3
4
5
5.0
5.2
5.4
5.6
5.8
6.0
6.2
Predicted values
glm(retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean + Race + G




```r
# Binned plot
x.2019 = predict(poisson2019)
y.2019 = residuals(poisson2019)
binnedplot(x.2019,y.2019)
```



Comparison between Poisson Regression and Linear Regression

Retention for Students Starting College in 2018

```
# Compare fitness of linear regression and poisson regression
# md.2018 for linear regression of 2018 retention
# poisson2018 for poisson regression of 2018 retention
anova(md.2018, poisson2018)
```

```
## Analysis of Variance Table
##
## Model 1: retention ~ Race + EverReceivedPromiseAward + CumulativeGPA +
##     AttendanceRate + Gender + SAT_Total + KeystoneMean + Num_CTE +
##     Race:EverReceivedPromiseAward
## Model 2: retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean +
##     Race + Gender + ELLStatus + IEPGroup + EconDisad + SAT_Total +
##     CumulativeGPA + MagnetInd + EverReceivedPromiseAward + QualifiedforCorePromise +
##     semester + Race * EverReceivedPromiseAward
##   Res.Df      RSS Df Sum of Sq     F    Pr(>F)
## 1    473 12986100
## 2    465    31222  8  12954877 24117 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Retention for Students Starting College in 2019

```r
# Compare fitness of linear regression and poisson regression
# md.2019 for linear regression of 2019 retention
# poisson2019 for poisson regression of 2019 retention
anova(md.2019, poisson2019)
```

```
## Analysis of Variance Table
##
## Model 1: retention ~ Race + EverReceivedPromiseAward + CumulativeGPA +
##     semester + SAT_Total + Gender + ELLStatus + Race:EverReceivedPromiseAward
## Model 2: retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean +
##     Race + Gender + ELLStatus + EconDisad + SAT_Total + CumulativeGPA +
##     MagnetInd + EverReceivedPromiseAward + QualifiedforCorePromise +
##     semester + Race:EverReceivedPromiseAward
##   Res.Df     RSS Df Sum of Sq     F    Pr(>F)
## 1    622 5711445
## 2    615   23250  7   5688195 21495 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusions:

- Since the p-value is very small ($< 2.2 \times 10^{-16}$), we say that poisson model fits better than linear regression model for 2018 retention.
- Since the p-value is also very small ($< 2.2 \times 10^{-16}$), we say that poisson model fits better than linear regression model for 2019 retention.
- Thus, we will employ poisson regression model for students' retention in college.

Part III: Multivariate Regression Analysis for Box Range

Data Wrangling

```
# Current GPA and attendance Cutoff: GPA 2.5, attendance 90%
# Create different cutoffs for GPA and attendance
gpa_upper = seq(2.9, 3.2, by = 0.1)
gpa_lower = seq(2.1, 1.8, by = -0.1)
attn_upper = seq(0.94, 1, by = 0.02)
attn_lower = seq(0.86,0.8, by = -0.02)
range_df = data.frame(attn_lower,attn_upper, gpa_lower,gpa_upper)
range_df
```

```
##   attn_lower attn_upper gpa_lower gpa_upper
## 1       0.86       0.94       2.1       2.9
## 2       0.84       0.96       2.0       3.0
## 3       0.82       0.98       1.9       3.1
## 4       0.80       1.00       1.8       3.2
```

```
# Clean data
# retention_reg data is the final data set used for multivariate regression analysis
# for whole range of GPA and attendance
retention_joint = read.csv("retention_reg.csv", stringsAsFactors = T)
retention_joint$RandomID <- as.character(retention_joint$RandomID)
retention_joint <- retention_joint %>%
  mutate(Gender = ifelse(Gender == "Female",1,0),
         ELLStatus = ifelse(ELLStatus == "ELL",1,0),
         EconDisad = ifelse(EconDisad == "Free Lunch",1,0),
         QualifiedforCorePromise = ifelse(QualifiedforCorePromise == "yes",1,0)) %>%
  dplyr::select(-c(X, AttendaceRateCate, GradYear, retention)) %>%
  inner_join(nsc_retention, by = c("RandomID","ENROLLMENT_BEGIN_year", "semester")) %>%
  mutate(retention = as.numeric(retention))
head(retention_joint)
```

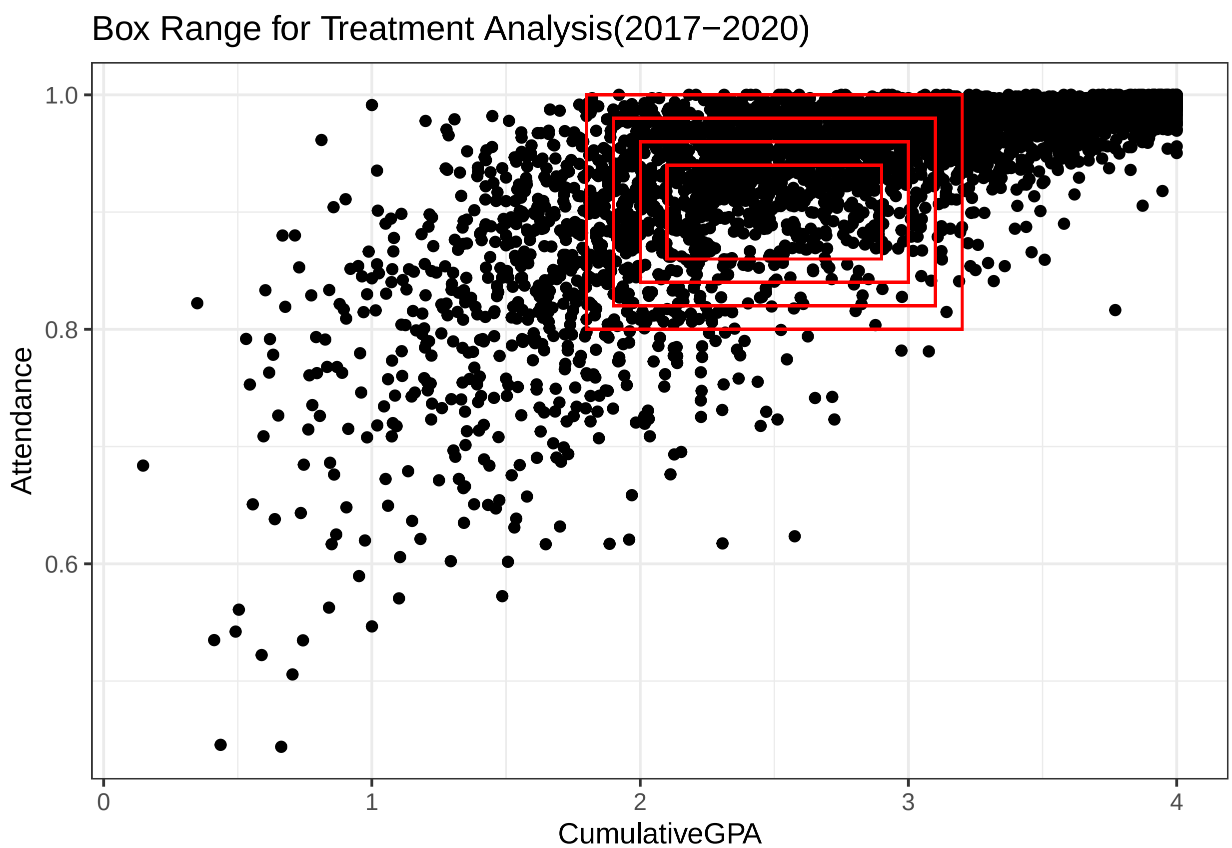
```
##   RandomID AttendanceRate Num_AP Num_CTE KeystoneMean             Race Gender
## 1  5841218      0.7552156      0       3     1435.357 African American      1
## 2  5849658      0.9944367      6       0     1516.250            White      0
## 3  5861017      0.9833102      0       0     1500.400 African American      0
## 4  5862641      0.9568846      0       0     1424.000 African American      0
## 5  5866142      0.9972184      3       0     1547.500            White      0
## 6  5869349      0.9860918      1       0     1464.167            White      1
##   ELLStatus          IEPGroup EconDisad SAT_Total CumulativeGPA MagnetInd
## 1         0               IEP         1       810         2.438         0
## 2         0 Not IEP or Gifted         0      1210         3.408         0
## 3         0 Not IEP or Gifted         1       970         2.117         1
## 4         0               IEP         1       740         2.856         0
## 5         0               IEP         1      1230         3.644         1
## 6         0               IEP         1       890         2.982         0
##   EverReceivedPromiseAward QualifiedforCorePromise ENROLLMENT_BEGIN_year
## 1                        1                       0                  2018
## 2                        1                       1                  2018
```

```
## 3                        0                        0                  2018
## 4                        1                        1                  2018
## 5                        1                        1                  2018
## 6                        1                        1                  2018
##   semester retention
## 1     Fall       217
## 2     Fall       548
## 3     Fall       337
## 4     Fall       544
## 5     Fall       563
## 6     Fall       724
```

```r
# Prepare data sets for box range analysis
# gpa_scholarship_attendance data is the joined data set of gpa, scholarship, and attendance
joint_df = read.csv("gpa_sholarship_attendance.csv")
names(joint_df)[11] <- "present_pct"
names(joint_df)[12] <- "present_pct_cate"
joint_df$RandomID <- as.character(joint_df$RandomID)

box_df = joint_df %>%
  dplyr::select(RandomID, CumulativeGPA, excused_pct) %>%
  inner_join(nsc_promise,  by = "RandomID") %>%
  filter(CumulativeGPA >=2, CumulativeGPA<=3) %>%
  filter(excused_pct >=0.85, excused_pct<=0.95)
```

EDA

```r
# Visualize the box ranges
ggplot(joint_df, aes(x =CumulativeGPA, y= excused_pct)) +
  geom_point() +
  labs(y = "Attendance") +
  geom_rect(aes(xmin = 2.1, xmax = 2.9, ymin = 0.86, ymax = 0.94), color = "red", alpha = 0)+
  geom_rect(aes(xmin = 2, xmax = 3, ymin = 0.84, ymax = 0.96), color = "red", alpha = 0)+
  geom_rect(aes(xmin = 1.9, xmax = 3.1, ymin = 0.82, ymax = 0.98), color = "red", alpha = 0)+
  geom_rect(aes(xmin = 1.8, xmax = 3.2, ymin = 0.8, ymax = 1), color = "red", alpha = 0) +
  theme_bw() +
  labs(title = "Box Range for Treatment Analysis(2017-2020)")
```




```
# Exclude semester and IEPGroup because of insufficient factor levels
get_box_df <- function(df, attn_lo, attn_high, gpa_lo, gpa_high){
  res = df %>%
    filter(AttendanceRate >=attn_lo, AttendanceRate<=attn_high) %>%
    filter(CumulativeGPA >=gpa_lo, CumulativeGPA<=gpa_high)
  return(res)
}
```

T-test for the treatment (EverReceivedPromiseAward)

```
# T-test for 2018
t.test.summary.2018 <- data.frame(estimate=numeric(),
                 lower=numeric(),
                 upper = numeric(),
                 pval=numeric(),
                 significant = logical())

# Conduct T-test for each box range
year = 2018
for(ind in 1:4){
  row = unlist(range_df[ind,])
  box_df = get_box_df(retention_joint,row[1], row[2],row[3],row[4])
  box_df_year = box_df %>% filter(ENROLLMENT_BEGIN_year == year)
  a = t.test(retention~EverReceivedPromiseAward, data = box_df_year)
```

```r
  res = data.frame(meanDiff=round(a$estimate[2]-a$estimate[1],3),
                   lower=round(a$conf.int[1],3),
                   upper = round(a$conf.int[2],3),
                   pval=round(a$p.value,3),
                   significant = a$p.value<=0.05)
  t.test.summary.2018 = rbind(t.test.summary.2018, res)

}
rownames(t.test.summary.2018) <- paste0("Box", 1:4)
t.test.summary.2018
```

```
##      meanDiff    lower   upper  pval significant
## Box1  145.118 -291.280   1.045 0.052       FALSE
## Box2  137.080 -228.130 -46.030 0.004        TRUE
## Box3  118.876 -201.006 -36.747 0.005        TRUE
## Box4  159.944 -223.736 -96.152 0.000        TRUE
```

```r
# T-test for 2019
t.test.summary.2019 <- data.frame(estimate=numeric(),
                   lower=numeric(),
                   upper = numeric(),
                   pval=numeric(),
                   significant = logical())

# Conduct T-test for each box range
year = 2019
for(ind in 1:4){
  row = unlist(range_df[ind,])
  box_df = get_box_df(retention_joint,row[1], row[2],row[3],row[4])
  box_df_year = box_df %>% filter(ENROLLMENT_BEGIN_year == year)
  a = t.test(retention~EverReceivedPromiseAward, data = box_df_year)
  res = data.frame(meanDiff=round(a$estimate[2]-a$estimate[1],3),
                   lower=round(a$conf.int[1],3),
                   upper = round(a$conf.int[2],3),
                   pval=round(a$p.value,3),
                   significant = a$p.value<=0.05)
  t.test.summary.2019 = rbind(t.test.summary.2019, res)

}
rownames(t.test.summary.2019) <- paste0("Box", 1:4)
t.test.summary.2019
```

```
##      meanDiff    lower   upper  pval significant
## Box1  110.344 -431.535 210.846 0.361       FALSE
## Box2   51.920 -164.177  60.337 0.327       FALSE
## Box3   61.896 -136.690  12.898 0.098       FALSE
## Box4   33.696  -84.802  17.409 0.188       FALSE
```

Findings:

- We see that for 2018, the treatment effect is significant for all boxes except for the first one. This is probably because of limited observations in the smallest box (24 observations).
- We see that for 2019, none of the boxes has significant treatment effect. This aligns with our box plot for the whole data set above.

Multivariate Linear Regression

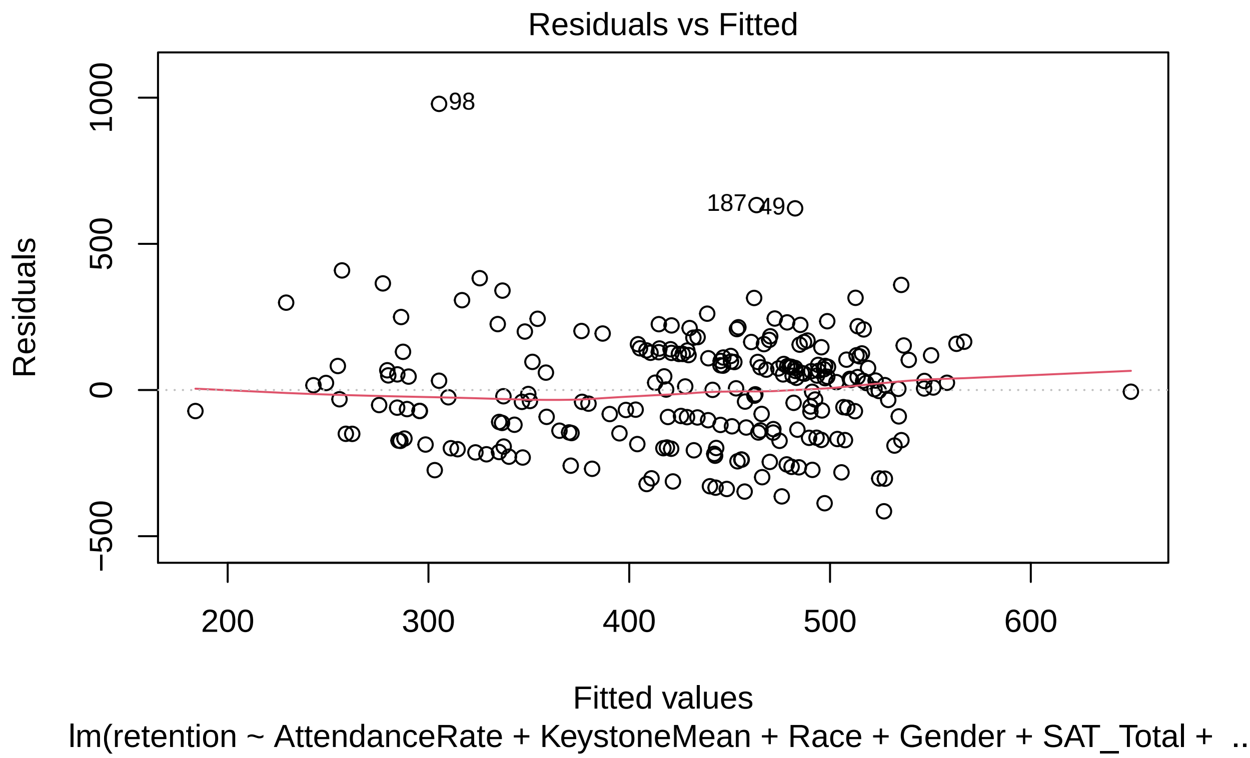
- We perform linear regression on retention with all other predicting variables.
- The linear regression serves as an extra step to verify the treatment effect of scholarship on retention.

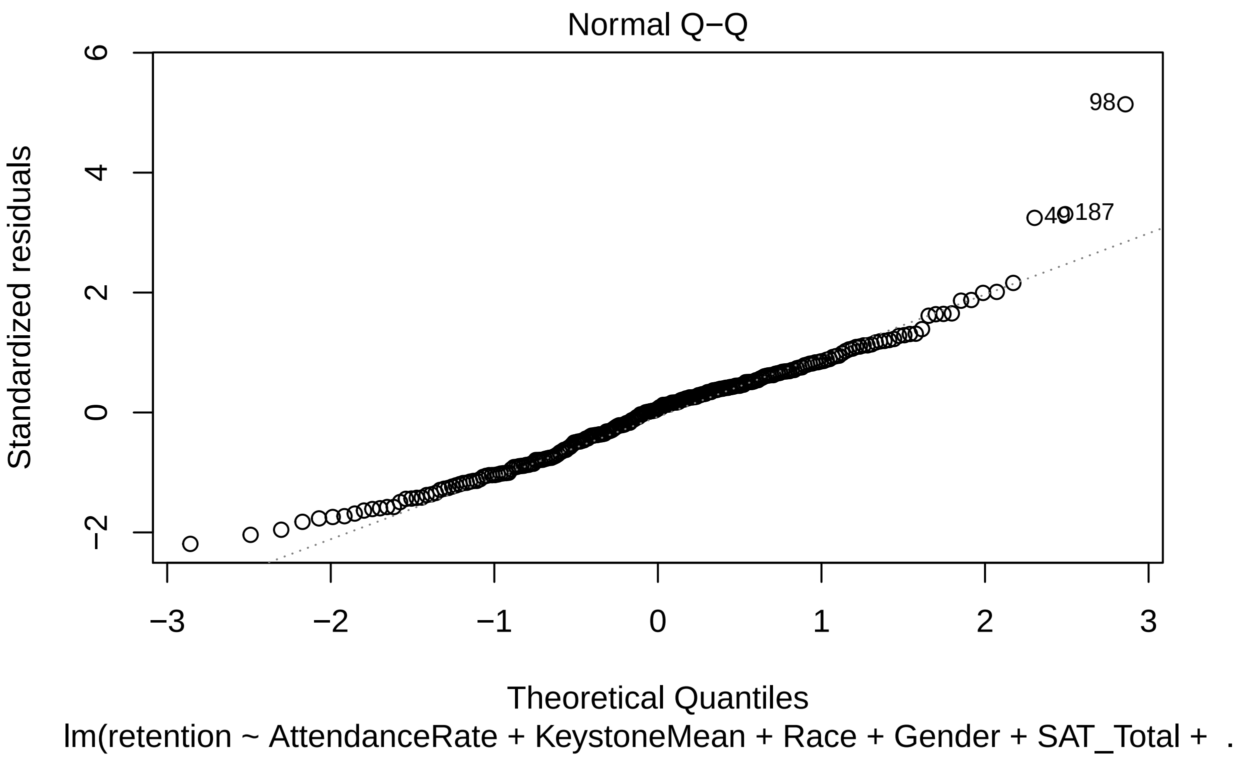
```r
# Define null model and full model for multivariate linear regression
form.0 = as.formula("retention ~ Race + EverReceivedPromiseAward + Race*EverReceivedPromiseAward")
form.full = as.formula("retention ~ AttendanceRate + Num_AP + Num_CTE + KeystoneMean +
                       Race + Gender + ELLStatus + EconDisad + SAT_Total + CumulativeGPA +
                       MagnetInd + EverReceivedPromiseAward + QualifiedforCorePromise +
                       Race*EverReceivedPromiseAward")

# Create a function box_yearly to perform multivariate linear regression for each box range
box_yearly <- function(range_ind, year = 2018){
  row = unlist(range_df[range_ind,])
  box_df = get_box_df(retention_joint,row[1], row[2],row[3],row[4])
  box_df_year = box_df %>% filter(ENROLLMENT_BEGIN_year == year)
  print(paste("Number of students:", nrow(box_df_year)))
  lm.full.year = lm(form.full, data = box_df_year)
  res = step(lm.full.year, list(lower = form.0, upper = form.full), direction = "both", trace = 0)
  return(res)
}
```

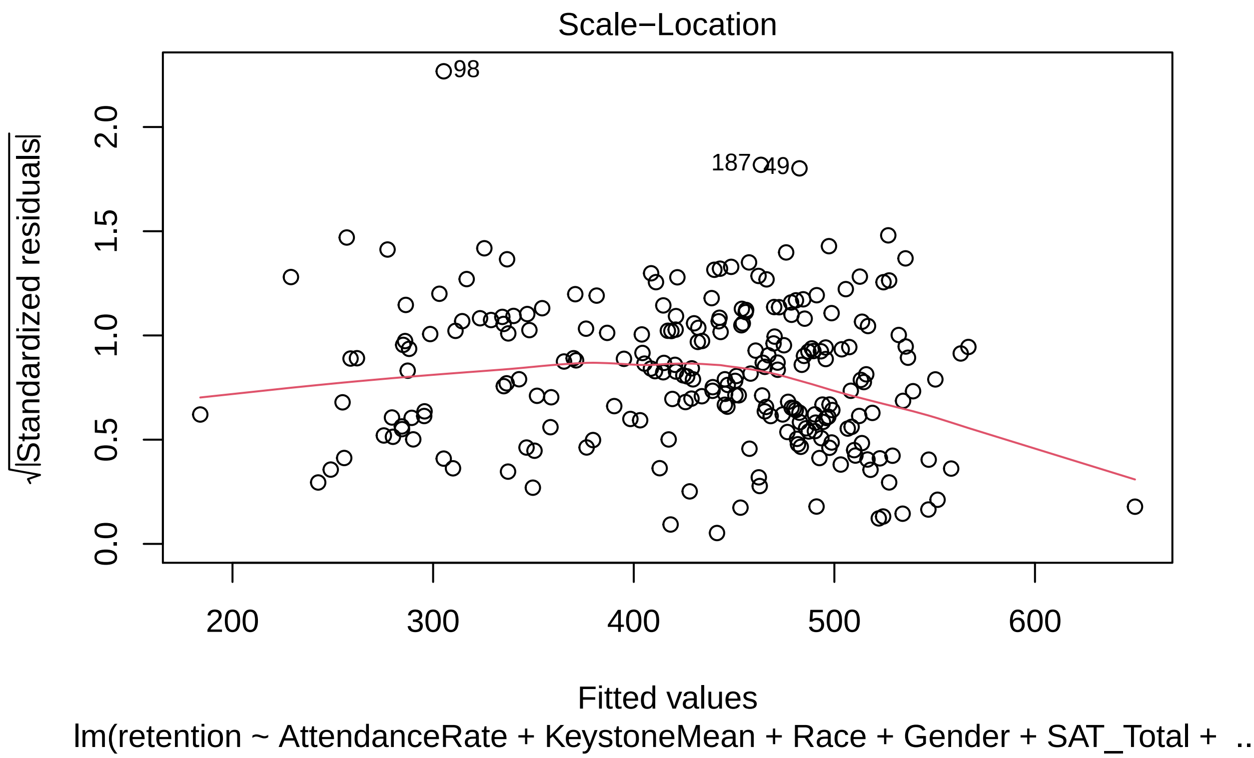
```r
# Construct linear regression model for each box range of 2018 retention
res.2018 = lapply(1:4, box_yearly)
```

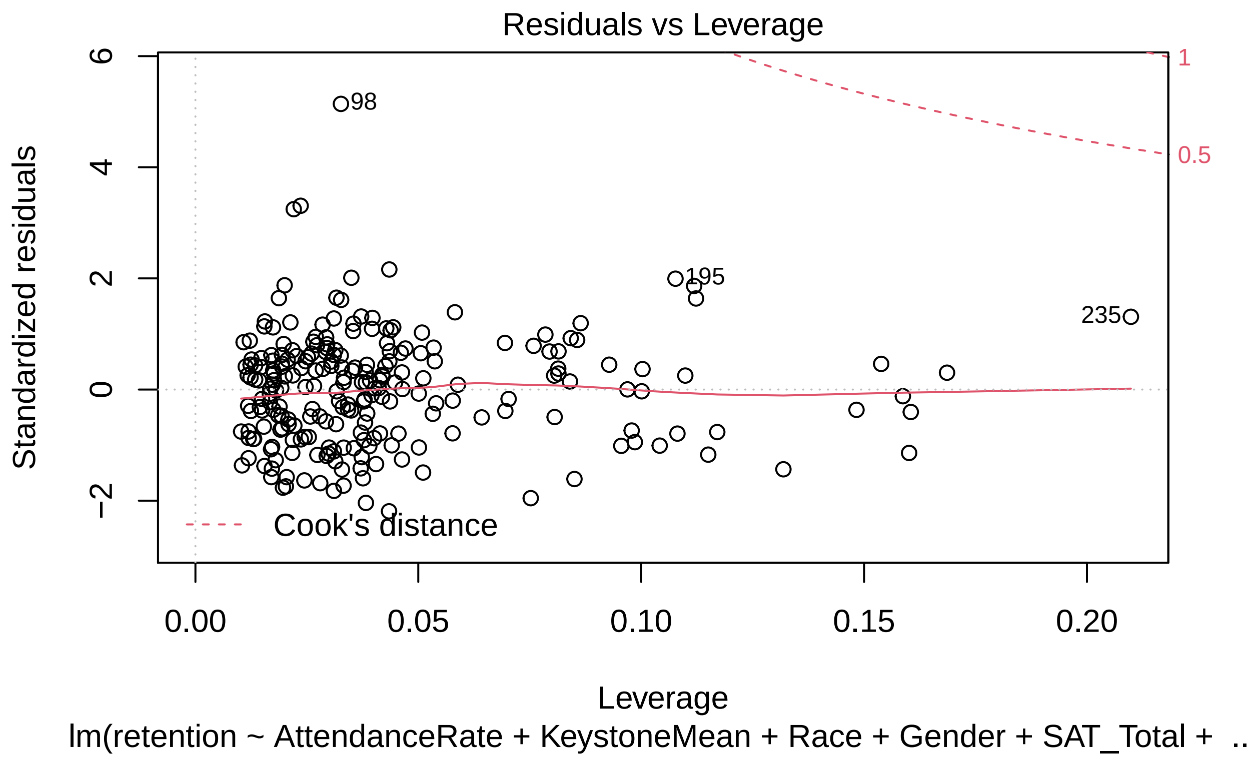
```
## [1] "Number of students: 34"
## [1] "Number of students: 75"
## [1] "Number of students: 150"
## [1] "Number of students: 235"
```

```r
# Model diagnostics for the last box
plot(res.2018[[4]])
```


Residuals vs Fitted
Residuals
Fitted values
1000
500
0
−500
200
300
400
500
600
98
187
49
lm(retention ~ AttendanceRate + KeystoneMean + Race + Gender + SAT_Total + ..


Normal Q−Q
Standardized residuals
6
4
2
0
−2
98
49
187
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(retention ~ AttendanceRate + KeystoneMean + Race + Gender + SAT_Total + ..


Scale−Location
98
187
49
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
200
300
400
500
600
Fitted values
lm(retention ~ AttendanceRate + KeystoneMean + Race + Gender + SAT_Total + ..



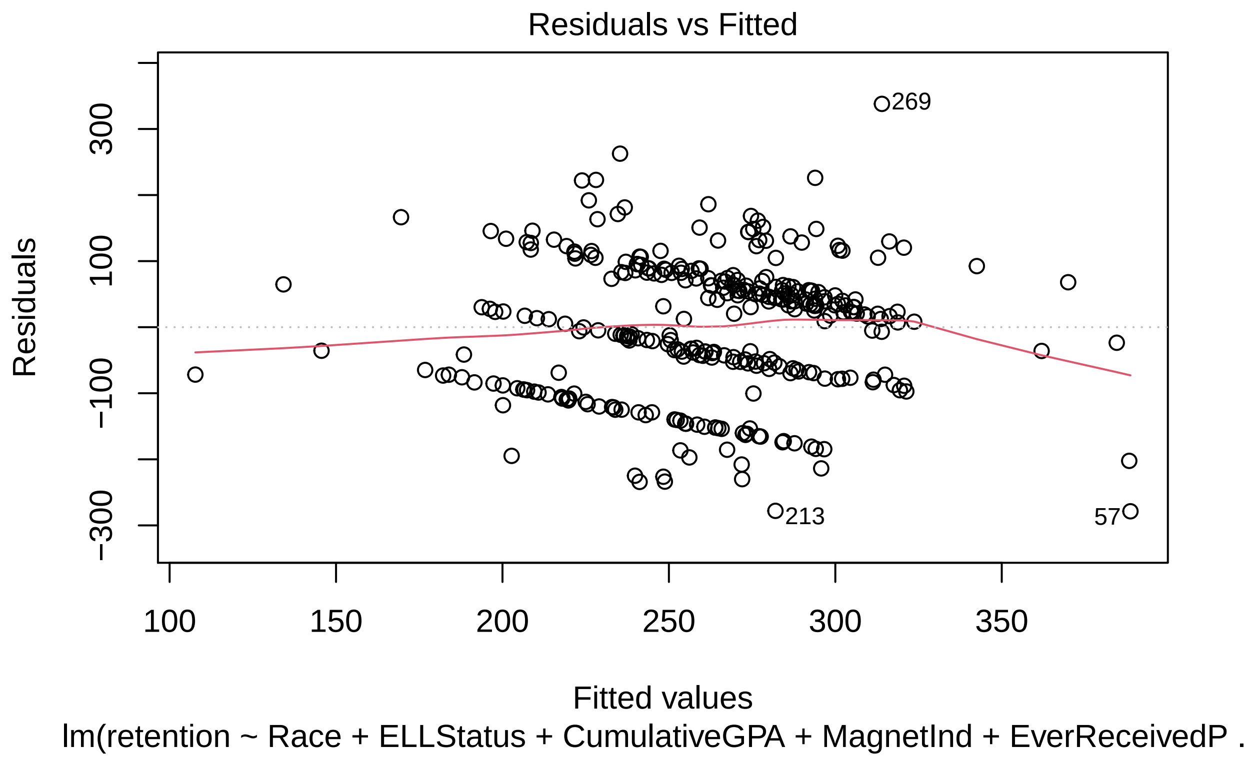

```r
# Coefficient output for the last box
(round((summary(res.2018[[4]]))$coef[,"Pr(>|t|)"],3))
```

```
##                          (Intercept)                       AttendanceRate
##                                0.370                                0.063
##                         KeystoneMean                            RaceOther
##                                0.084                                0.849
##                            RaceWhite                               Gender
##                                0.613                                0.047
##                            SAT_Total             EverReceivedPromiseAward
##                                0.013                                0.000
## RaceOther:EverReceivedPromiseAward RaceWhite:EverReceivedPromiseAward
##                                0.801                                0.562
```

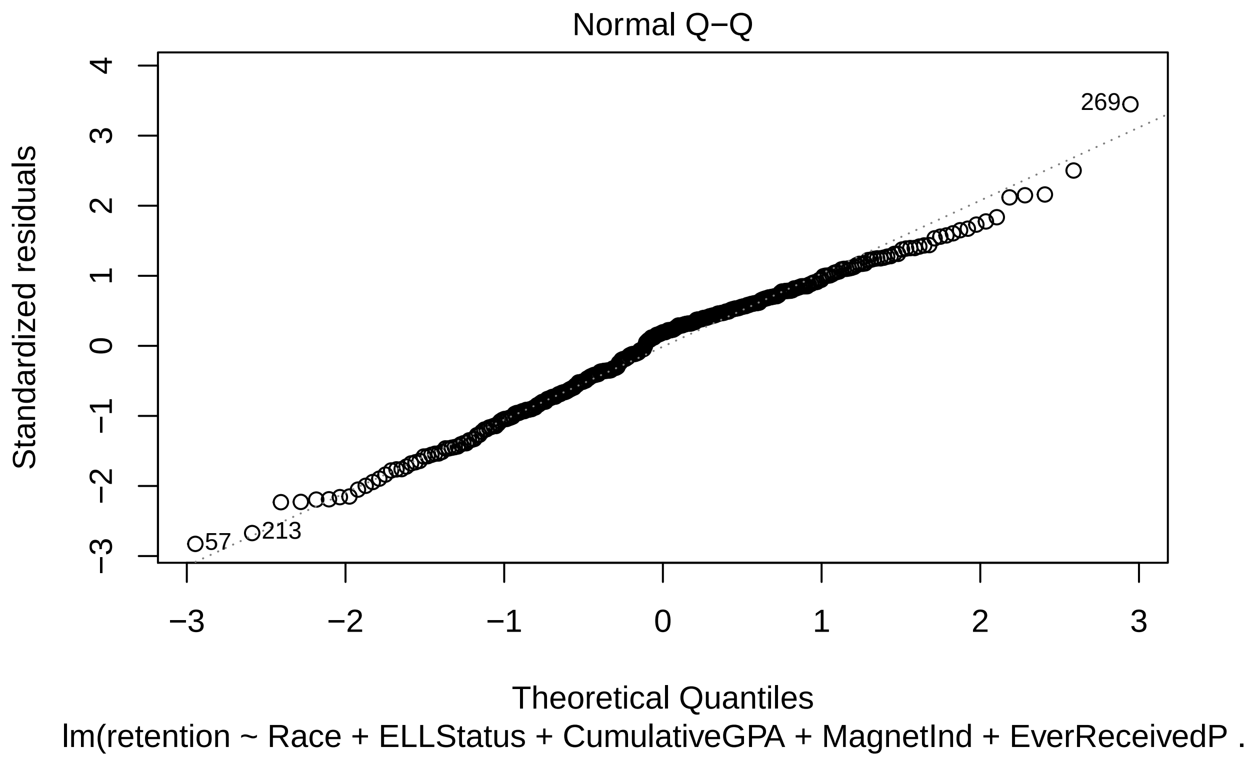
```r
# Construct linear regression model for each box range of 2019 retention
res.2019 = lapply(1:4, box_yearly, year = 2019)
```

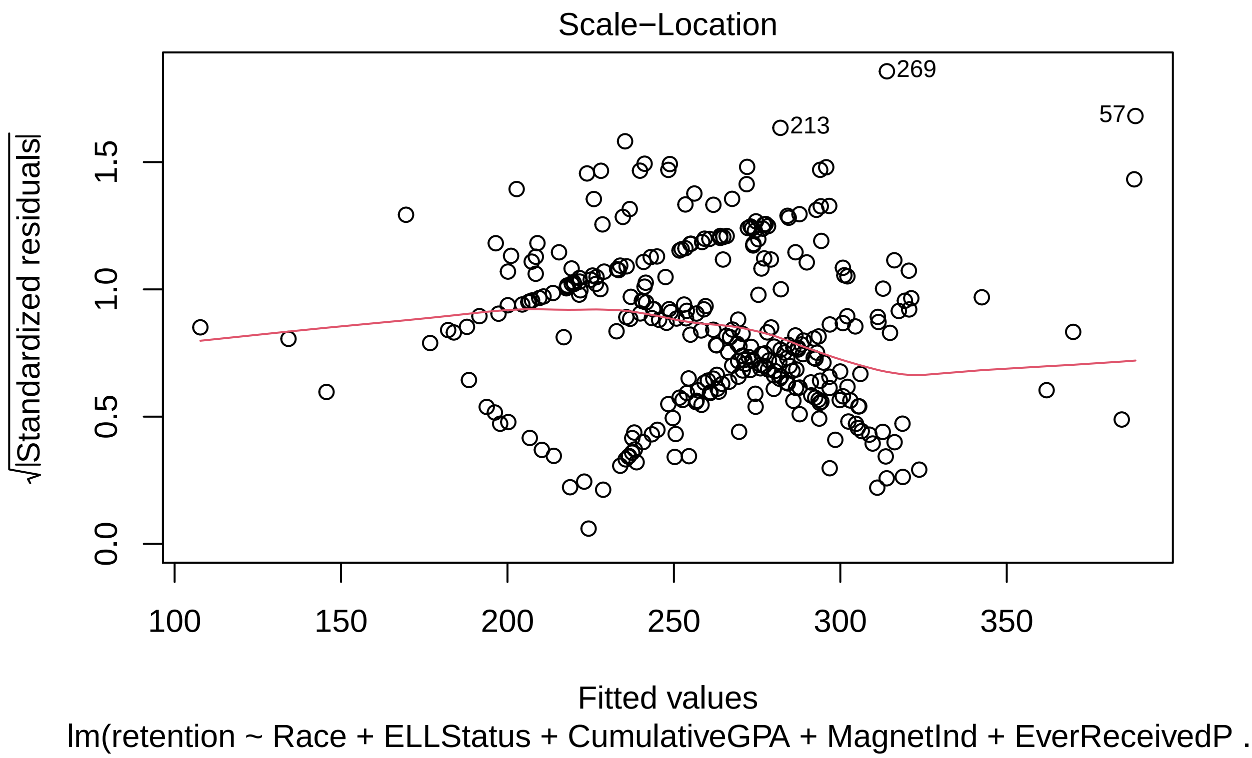
```
## [1] "Number of students: 57"
## [1] "Number of students: 110"
## [1] "Number of students: 203"
## [1] "Number of students: 311"
```

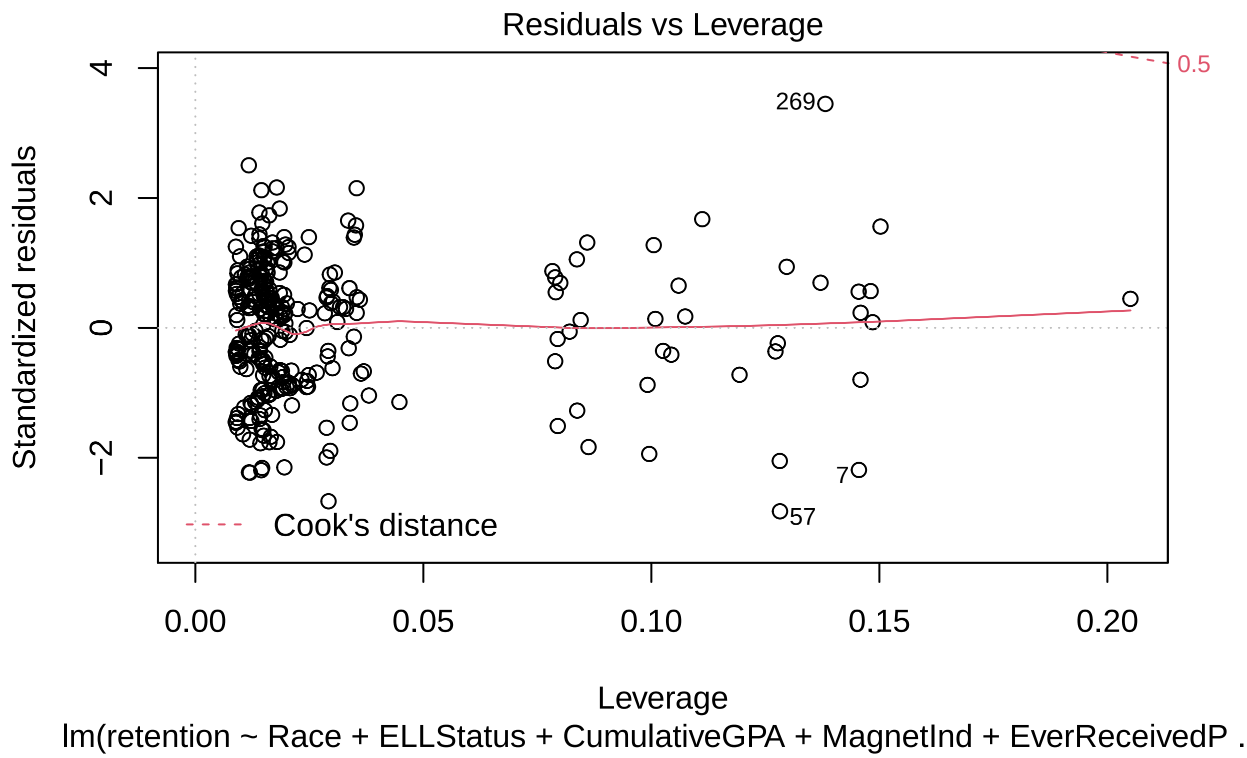
```r
# Model diagnostics for the last box
plot(res.2019[[4]])
```



Residuals vs Fitted
269
213
57
300
100
−100
−300
Residuals
100
150
200
250
300
350
Fitted values
lm(retention ~ Race + ELLStatus + CumulativeGPA + MagnetInd + EverReceivedP .


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(retention ~ Race + ELLStatus + CumulativeGPA + MagnetInd + EverReceivedP .
269
57
213
−3
−2
−1
0
1
2
3
4


Scale−Location
269
57
213
√|Standardized residuals|
0.0
0.5
1.0
1.5
100
150
200
250
300
350
Fitted values
lm(retention ~ Race + ELLStatus + CumulativeGPA + MagnetInd + EverReceivedP .




```r
# Coefficient output for the last box
(round((summary(res.2019[[4]]))$coef[,"Pr(>|t|)"],3))
```

```
##                         (Intercept)                           RaceOther
##                               0.919                               0.785
##                           RaceWhite                           ELLStatus
##                               0.840                               0.054
##                       CumulativeGPA                           MagnetInd
##                               0.000                               0.054
##             EverReceivedPromiseAward RaceOther:EverReceivedPromiseAward
##                               0.381                               0.026
## RaceWhite:EverReceivedPromiseAward
##                               0.419
```

Notice:

- For simplicity, we only demonstrate the model diagnosis for the last box.
- Replacing the index "4" with 1 through 3 to see model performance and coefficients for other 3 smaller boxes.
- More results of the linear regression will be discussed in the paper.