Analysis of FIRST Robotics Team Performance

SDS 230 – Spring 2022

4/29/2022

Introduction

FIRST Robotics Competitions (FRC) is a national organization that runs robotics teams in over 2500 high schools across the United States. My robotics team motivated me to study engineering, but more broadly they have a huge impact in encouraging students to study STEM. This report will investigate inequities in the program by analyzing the correlation of various socio-economic indicators with team performance. This will be done using socioeconomic and 2022 team performance data retrieved from Census.gov and The Blue Alliance API respectively. The hope is that a better understanding of how socio-economics impacts robotics teams can help guide equity programs to correct inequalities.

Data

Variables

- *numYears*: The total number of years a given high school robotics team has been active (2022 data). (discrete var.)
- *numAwards*: The total number of competition awards a given team won in the 2022 season. (discrete var.)
- *wasAtCMP*: An indicator that's TRUE only if a given team competed at the 2022 FRC World Championship. (categorical indicator var.)
- *ccmp*: The calculated contribution to winning margin (CCWM) quantifies the strength of a team's robot (unitless, but higher indicates a better robot). Data for this variable is only available for the teams who competed at the World Championship. (continuous var.)
- *State_prov*: The US State from which a given team is based. (categorical var.)
- *region*: Census Bureau-designated regional location of given team's state (e.g. Northeast). (categorical var.)
- *median_income*: The 2021 median household income in the zip code of a given team's high school (2020 inflation-adjusted $). (continuous var.)
- *higher_ed_perc*: The percent of adults (age 25+) with a college degree in the zip code of a given team's high school (2020 data). (continuous var.)
- *pop_dens*: The population density (persons per square mile) in the zip code of a given team's high school. (continuous var.)

Data Scraping and Cleaning

Robotics Data

An organization called The Blue Alliance (https://www.thebluealliance.com/apidocs) hosts a database of team

data, only accessible through their API. Learning how to execute authenticated API queries, then translating the JSON response into an R data frame was a significant challenge. Then, for each variable of interest, I wrote a loop to systematically retrieve data for it. Finally, I merged the data into a single data frame. For the sake of brevity, the scraping output has been omitted from this final report, although all of the code (except a personal authentication key used to access the data) has been included below to demonstrate the process

Census Data

On the Census Website (https://data.census.gov/cedsci/), I found databases containing income and education data by zip code (ZCTA). One challenge was the Census website being confusing to navigate. Cleaning the databases involved eliminating variables I wasn't interested in, and converting character values into numeric ones. The educational attainment data was given in number of people, so I manually made the data into a percent. Combining the robotics data and census data required merging the two dataframes by zip code which is the variable they shared. Finally, I merged in population density data from source (https://dashboards.securedatakit.com/public/dashboard/5b8fe700-531b-416b-9cf5-3ee9b93a2353), and organized the teams into regional categories.

R Code

```
library(rjson) # for JSON translation
library(httr)  # for API Queries
```

```
# changing these 2 parameters allows for easily re-running the analysis
# on data from a different year
year <- 2022
chmp_div_codes_2022 <- c("2022carv", "2022gal", "2022hop",
                         "2022new", "2022roe", "2022tur", "2022cmptx")
```

```
# query The Blue Alliance API: https://www.thebluealliance.com/apidocs/v3
# for general info on teams that participated in the 2022 FRC robotics season

baseaddress <- paste0("https://www.thebluealliance.com/api/v3/teams/", year, "/")
queryaddress <- paste0(baseaddress, 0)

page_result <- queryAndconvertJSON(queryaddress) # retrieves data and stores as list
team_data <- fixNullandMakeDf(page_result)       # converts list to data frame

# continue querying pages until all data has been scraped (~17 pages)
page <- 1
while(TRUE) {
  queryaddress <- paste0(baseaddress, page)
  page_result <- queryAndconvertJSON(queryaddress)

  if (length(page_result)==0) break # all data has been scraped

  page_team_data <- fixNullandMakeDf(page_result)

  team_data <- rbind(team_data, page_team_data)
  page <- page + 1
}

# reduces data set to US teams, as to make census data relevant
team_data <- team_data[team_data$country == "USA", ]
```

```
# query the blue alliance API for number of years each team has been active,
# and the number of awards won by each time in the {year} season

baseaddress <- "https://www.thebluealliance.com/api/v3/team/"

numYears = rep(NA, length(team_data$key))
numAwards = rep(NA, length(team_data$key))

for (i in 1:length(team_data$key)) {
  team_key <- team_data$key[i]

  # number of years active
  years_participated_queryaddress <- paste0(baseaddress, team_key, "/years_participated")
  numYears[i] <- length(queryAndconvertJSON(years_participated_queryaddress))

  # number of awards won
  awards_queryaddress <- paste0(baseaddress, team_key, "/awards/", year)
  numAwards[i] <- length(queryAndconvertJSON(awards_queryaddress))
  }

team_data <- cbind(team_data, numYears)
team_data <- cbind(team_data, numAwards)
```

```
# adds a variable to the dataset indicating whether each team qualified attended
# the 2022 world championship (TRUE indicates that the team did attend)

baseaddress <- "https://www.thebluealliance.com/api/v3/event/"

teams_at_championship = c()

# FRC championship 2022 Was broken up into 7 divisions
# combine the list of teams at any of the divisions
for (event_key in chmp_div_codes_2022) {
  division_queryaddress <- paste0(baseaddress, event_key, "/teams/keys")
  teams_in_division <- queryAndconvertJSON(division_queryaddress)

  teams_at_championship <- union(teams_at_championship, teams_in_division)
}

wasAtCMP <- (team_data$key %in% teams_at_championship)
team_data <- cbind(team_data, wasAtCMP)
```

```
# For each team that attended the world championships, adds a variable indicating how their
# robot performed as measured by its calculated contributions to winning margin (CCWM)

baseaddress <- "https://www.thebluealliance.com/api/v3/event/"

CCWM_data <- NA

# FRC championship 2022 Was broken up into 7 divisions
# combine the list of teams at any of the divisions
for (event_key in chmp_div_codes_2022) {
  CCWM_queryaddress <- paste0(baseaddress, event_key, "/oprs")

  ccwm_raw <- queryAndconvertJSON(CCWM_queryaddress)$ccwms
  ccwm_df_temp <- data.frame("key" = names(ccwm_raw), "ccwm" = unlist(ccwm_raw))

  CCWM_data <- rbind(CCWM_data, ccwm_df_temp)[-1, ]
}

team_data <- merge(team_data, CCWM_data, by = "key", all.x = TRUE)
```

```
# read in and clean census demographic data on level of education
# calculate the % of residents in each zip code with a college degree

educ_attainment <- read.csv("educational_attainment_2020_USCensus_B15003.csv")

colnames(educ_attainment) <- educ_attainment[1,]
educ_attainment <- educ_attainment[-1,]

zip <- educ_attainment[[c(52)]]
zip <- gsub("^.* ", "", zip)

total <- educ_attainment[[c(1)]]
total <- as.numeric(total)

higher_ed <- educ_attainment[, c(seq(41, 49, 2))]
higher_ed <- as.data.frame(lapply(higher_ed, as.numeric))
total_higher_ed <- apply(higher_ed, 1, sum)

percent_higher_ed <- total_higher_ed / total * 100

cleaned_ed_data <- data.frame(postal_code = zip, higher_ed_perc = percent_higher_ed)
team_data <- merge(team_data, cleaned_ed_data, by = 'postal_code', all.x = TRUE)
```

```
# read in and clean census demographic data on household income by zip code

income_data <- read.csv("income_past12months_USCensus_S1901.csv")

colnames(income_data) <- income_data[1,]
income_data <- income_data[-1,]

zip <- income_data$`Geographic Area Name`
zip <- gsub("^.* ", "", zip)

median_income <- income_data$`Estimate!!Families!!Median income (dollars)`
median_income <- as.numeric(median_income)

cleaned_income_data <- data.frame(postal_code = zip, median_income = median_income)
team_data <- merge(team_data, cleaned_income_data, by = 'postal_code', all.x = TRUE)
```

```
# import population density by zip code and merge with main dataset

pop_data <- read.csv("us_population_density_by_zip_september2020.csv")

pop_density <- pop_data[c("zip", "population_density")]
names(pop_density) <- c("postal_code", "pop_dens")

team_data$postal_code <- as.numeric(gsub("-.*$", "", team_data$postal_code))
team_data <- merge(team_data, pop_density, by = 'postal_code', all.x = TRUE)
```

```
# group states into regional categories

West <- c("Arizona", "Colorado", "Idaho", "Montana", "Nevada", "New Mexico", "Utah", "Wyomin
g", "Alaska", "California", "Hawaii", "Oregon", "Washington")
Midwest <- c("Illinois", "Indiana", "Michigan", "Ohio", "Wisconsin", "Iowa", "Kansas", "Minne
sota", "Missouri", "Nebraska", "North Dakota", "South Dakota")
South <- c("Delaware", "Florida", "Georgia", "Maryland", "North Carolina", "South Carolina",
"Virginia", "District of Columbia", "West Virginia", "Alabama", "Kentucky", "Mississippi", "T
ennessee", "Arkansas", "Louisiana", "Oklahoma", "Texas")
Northeast <- c("Connecticut", "Maine", "Massachusetts", "New Hampshire", "Rhode Island", "Ver
mont", "New Jersey", "New York", "Pennsylvania")

region <- rep(NA, length(data$state_prov))

for (i in 1:length(data$state_prov)) {
  regTemp <- NA
  state <- data$state_prov[i]

  if (state %in% West){
    regTemp <- "West"
  } else if (state %in% Midwest) {
    regTemp <- "Midwest"
  } else if (state %in% South) {
    regTemp <- "South"
  } else if (state %in% Northeast) {
    regTemp <- "Northeast"
  }

  region[i] <- regTemp
}

team_data <- cbind(team_data, region)
```

```
# save the scraped data locally to avoid having to repeat all the API calls when reloading R
save(team_data, file = "team_data.RData")
```

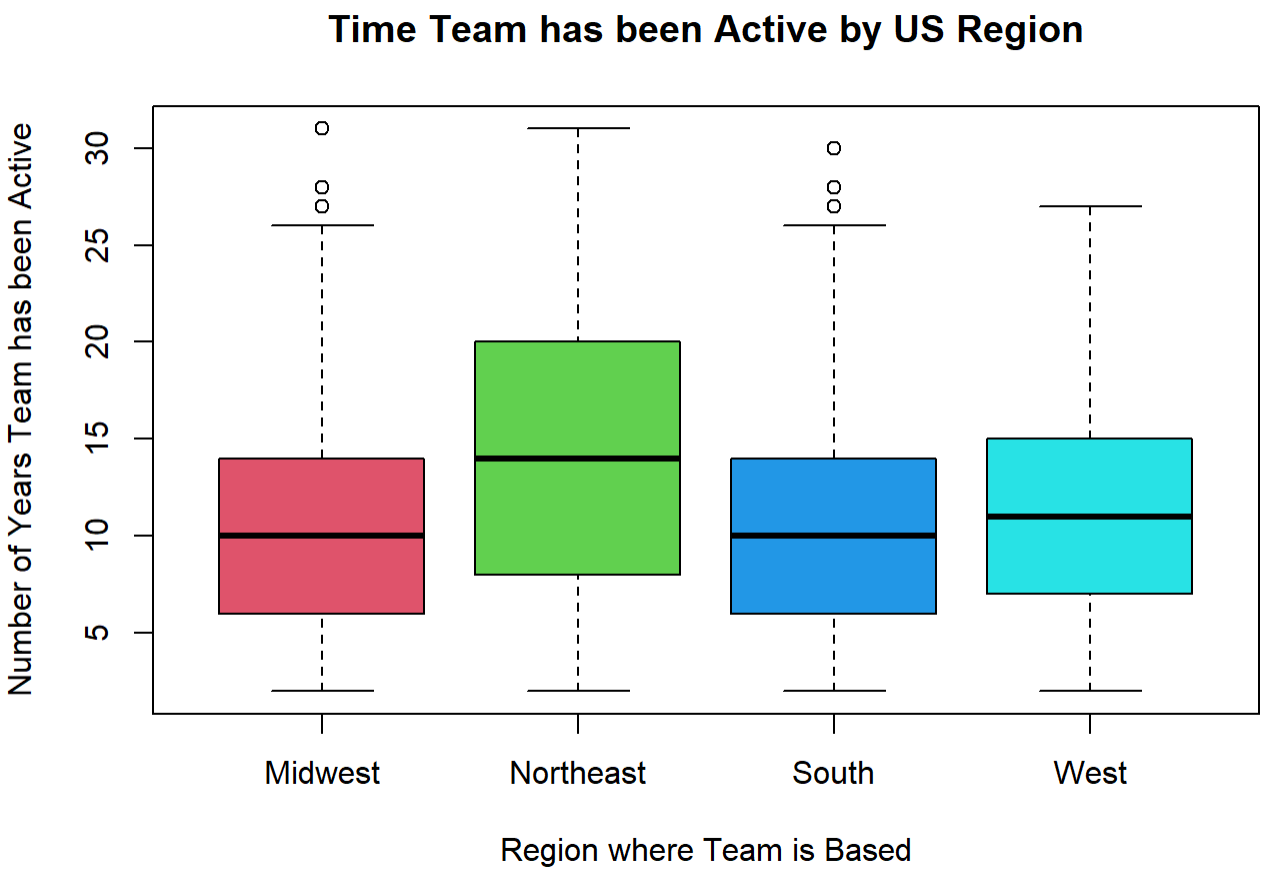
```
# The scraped dataset used null to indicate missing values; this code replaces those nulls wi
th NAs
# then restructures R's interpretation of the original JSON file into a dataframe
fixNullandMakeDf <- function(list1) {
  df = as.data.frame(lapply(list1[[1]], function(x) ifelse(is.null(x), NA, x)))

  # iterates through a list of lists where each sublist was a row of the original JSON
  for (i in 2:length(list1)) {
    new_row <- lapply(list1[[i]], function(x) ifelse(is.null(x), NA, x))  # replaces nulls wi
th NAs

    df <- rbind(df, as.data.frame(new_row))
  }

  df
}

# Given a web address in TBA API, and converts JSON to list
queryAndconvertJSON <- function(webaddress) {
  auth_key <- "" # place personal authentication key. www.thebluealliance.com/apidocs
  r <-GET(webaddress, add_headers("X-TBA-Auth-Key" = auth_key))

  fromJSON(rawToChar(r$content))
}
```

```
# load the scraped data that's been stored locally
load("team_data.RData")

data <- team_data[c("team_number", "city", "state_prov", "postal_code", "region",
                    "numYears", "numAwards", "ccwm", "wasAtCMP", "median_income",
                    "higher_ed_perc", "pop_dens")]

data <- data[complete.cases(data[, -8]), ]

head(data, 6)
```

```
##   team_number     city state_prov postal_code  region numYears numAwards ccwm
## 1           1  Pontiac   Michigan       48340 Midwest       26         1   NA
## 2         100 Woodside California       94062    West       26         0   NA
## 3        1002 Marietta    Georgia       30068   South       20         2   NA
## 4         101  Chicago   Illinois       60634 Midwest       26         0   NA
## 5        1011   Tucson    Arizona       85741    West       20         0   NA
## 6        1014   Dublin       Ohio       43017 Midwest       20         1   NA
##   wasAtCMP median_income higher_ed_perc pop_dens
## 1    FALSE         48687       18.29876  3322.53
## 2    FALSE        225684       72.50760   336.59
## 3    FALSE        154695       74.94993  2188.39
## 4    FALSE         85050       33.41722 10611.96
## 5    FALSE         69334       40.45991  3203.41
## 6    FALSE        133848       75.63727  1946.22
```

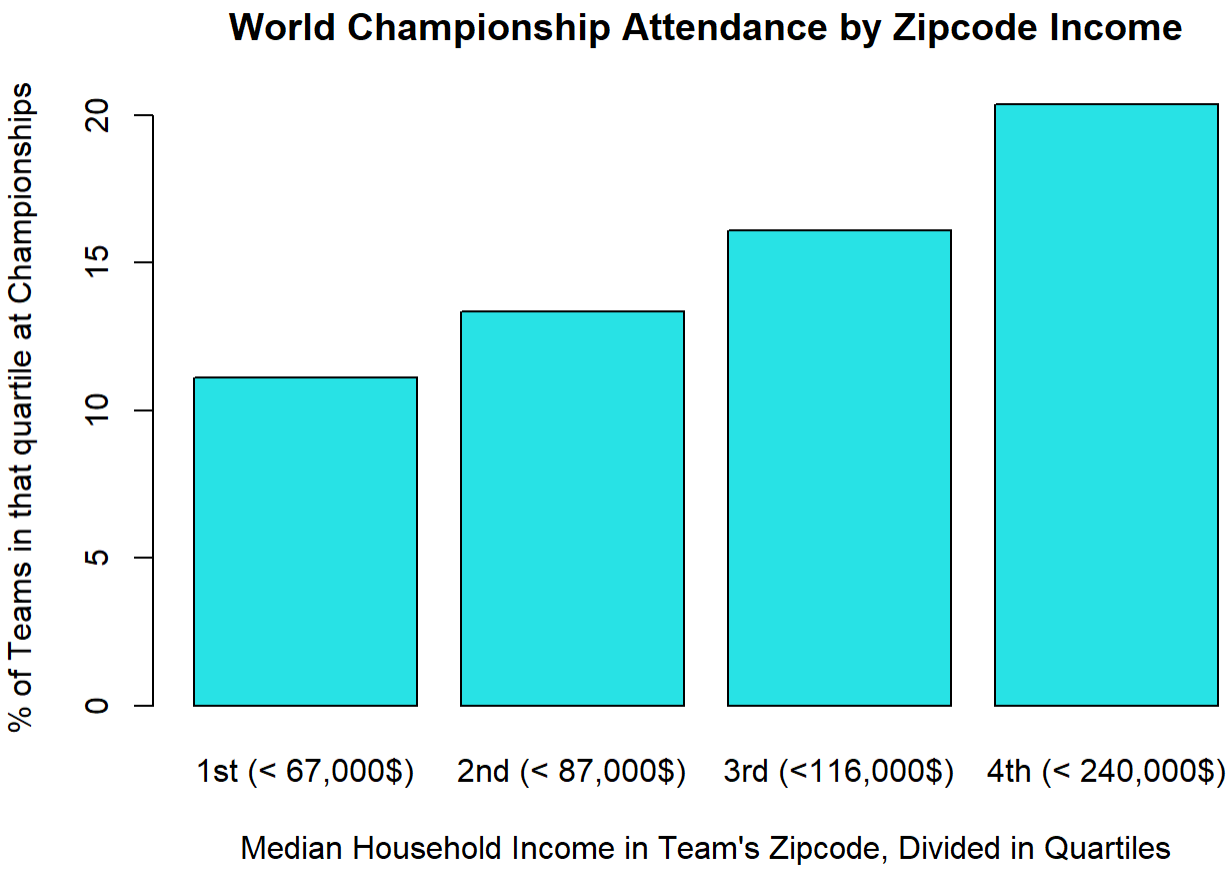
Descriptive Plots

```
boxplot(data$numYears ~ data$region, col = 2:5, xlab = "Region where Team is Based", ylab = "
Number of Years Team has been Active", main = "Time Team has been Active by US Region")
```



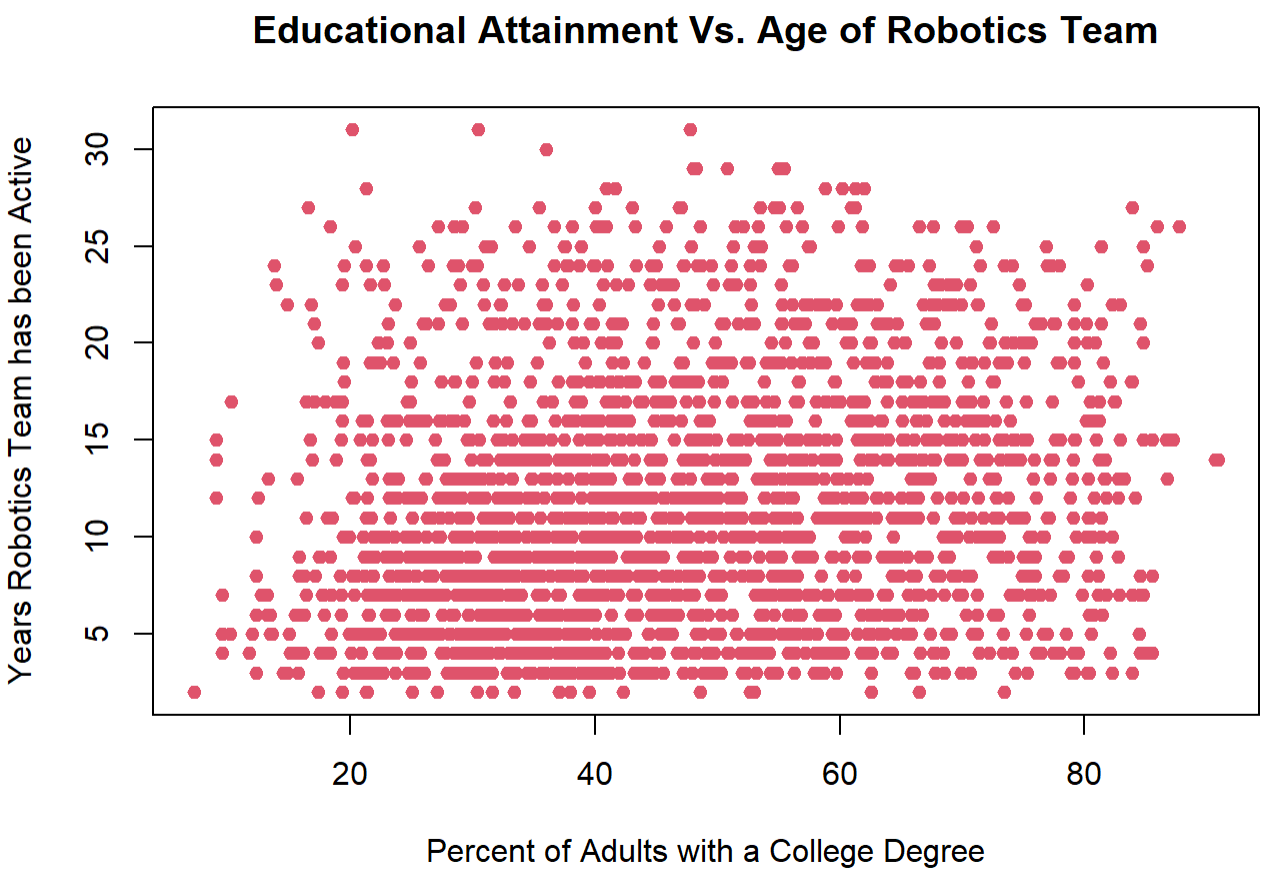
This boxplot shows that teams in the Northeast visually appear to have competed for longer on average than teams in the Midwest, South, and West. This is to be expected, because the FIRST organization began in the Northeast.

```
income_quantiles <- quantile(data$median_income)
perc_qual_from_income_quantile <- rep(NA, length(income_quantiles) - 1)

for (i in 1:(length(income_quantiles) - 1)) {
  perc_qual_from_income_quantile[i] <-
    nrow(data[(data$median_income > income_quantiles[i]) & (data$median_income < income_quant
iles[i + 1]) & (data$wasAtCMP == TRUE),]) /
    nrow(data[(data$median_income > income_quantiles[i]) & (data$median_income < income_quant
iles[i + 1]),]) * 100
}

income_quart_labels <- c("1st (< 67,000$)", "2nd (< 87,000$)", "3rd (<116,000$)", "4th (< 24
0,000$)")

barplot(perc_qual_from_income_quantile, names.arg = income_quart_labels, main = "World Champi
onship Attendance by Zipcode Income", xlab = "Median Household Income in Team's Zipcode, Divi
ded in Quartiles", ylab = "% of Teams in that quartile at Championships", col = 5)
```



From this histogram one can see that the higher the income bracket, the more teams from the bracket will be represented at world's. From the 4th and highest quartile just about a fifth of teams attend world's where only a little more than a tenth of teams from the lowest quartile of median income by zip code attend. There appears to be a positive association between median income and attendence at the world championship.

```
plot(data$higher_ed_perc, data$numYears, col = 2, pch = 19, cex = 0.9, xlab = "Percent of Adu
lts with a College Degree", ylab ="Years Robotics Team has been Active", main =  "Educational
Attainment Vs. Age of Robotics Team")
```



Educational attainment and age of robotics team seem to have a slight positive correlation, although it is hard to tell visually. A statistical test of this correlation is conducted in the analysis section of the report, and shows that the correlation is indeed significant.

Analysis

T-Test

```
# sourced from same census dataset (S1901)
US_overall_median_income <- 64994

t.test(data$median_income, mu = US_overall_median_income,
       alternative = "greater", conf.level = 0.99)
```

```
##
##  One Sample t-test
##
## data:  data$median_income
## t = 37.779, df = 2338, p-value < 2.2e-16
## alternative hypothesis: true mean is greater than 64994
## 99 percent confidence interval:
##  93156.58      Inf
## sample estimates:
## mean of x
##  95005.91
```

H0: The median household income where robotics teams are based is no different than the national average
Ha: Robotics teams are based in zipcodes with median household incomes higher than the national average

At a 99% confidence level, there is statistically significant evidence to reject the null hypothesis, suggesting robotics teams tend to be based in wealthier zipcodes.

Correlation

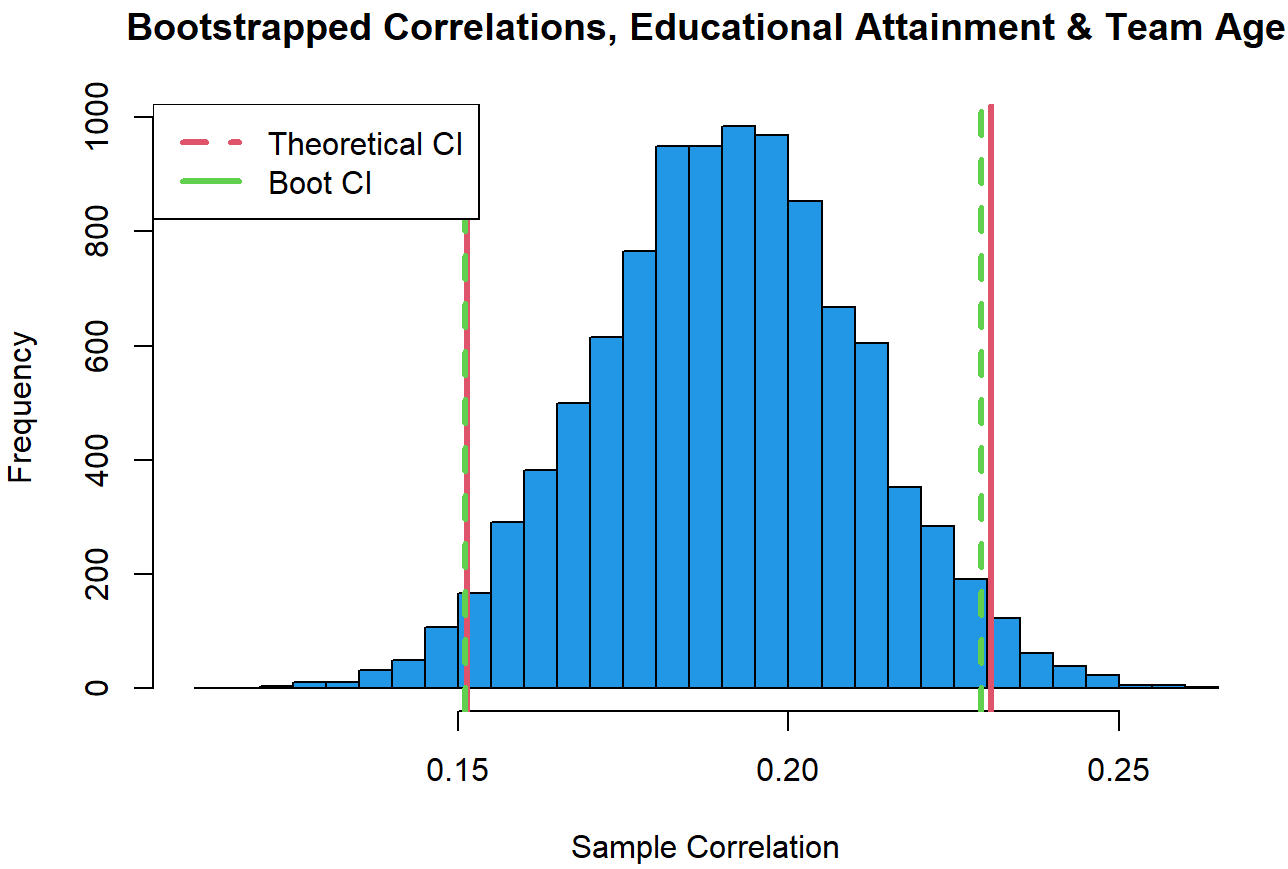
```
ctest1 <- cor.test(data$higher_ed_perc, data$numYears)
ctest1
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$higher_ed_perc and data$numYears
## t = 9.3748, df = 2337, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1510137 0.2291399
## sample estimates:
##       cor
## 0.1903782
```

There is a statistically significant correlation between the percent of adults in a zip code with a college degree, and the number of years a robotics team in that zipcode has been active. This is indicated by the 2.2e-16 p-value which is far below most reasonable alpha levels.

Bootstrap

```
higher_ed_perc <- data$higher_ed_perc
numYears <- data$numYears

N <- length(numYears)

n_samp <- 10000

corResults <- rep(NA, n_samp)

for (i in 1:n_samp) {
  # get the bootstrapped sample
  s <- sample(1:N, N, replace = T)
  fakeData <- data[s, ]

  # get bootstrapped correlation
  corResults[i] <- cor(higher_ed_perc[s], numYears[s])
}

# bootstrapped correlation 95% CI
ci_r <- quantile(corResults, c(0.025, 0.975))

hist(corResults, col = 4, main = "Bootstrapped Correlations, Educational Attainment & Team Ag
e", xlab = "Sample Correlation", breaks = 50)

abline(v = ci_r, lwd = 3, col = 2)
abline(v = ctest1$conf.int, lwd = 3, col = 3, lty = 2)
legend("topleft", c("Theoretical CI", "Boot CI"), lwd = 3, col = c(2, 3), lty = c(2,1))
```




```
ci_r
```

```
##      2.5%     97.5%
## 0.1513131 0.2305794
```

```
ctest1$conf.int
```

```
## [1] 0.1510137 0.2291399
## attr(,"conf.level")
## [1] 0.95
```

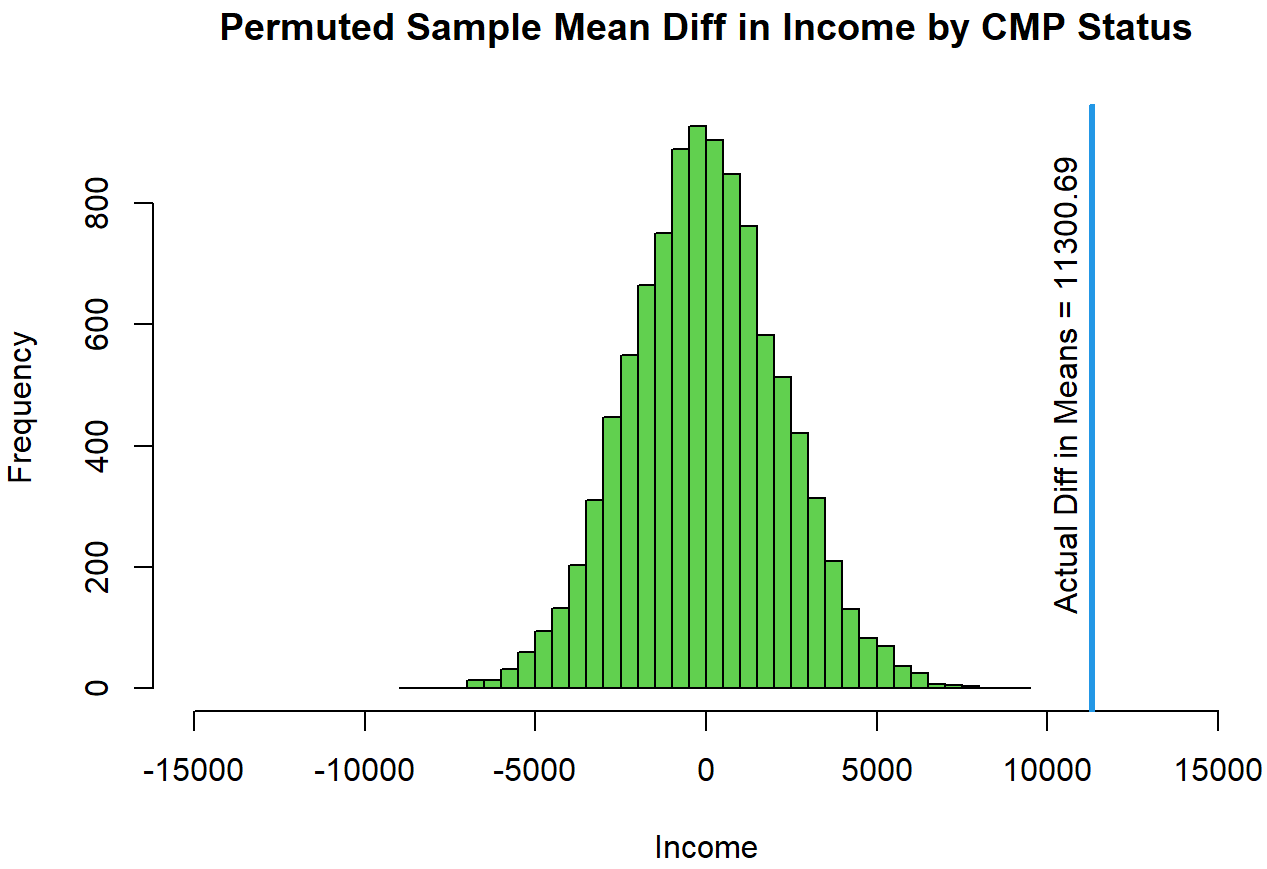
The 95% bootstrapped and theoretical confidence intervals for the true correlation between the percent of adults in a zip code with a college degree, and the number of years a robotics team in that zipcode has been active match almost exactly. The CI interval range is approximately 0.15 to 0.23.

Permutation Test

```
t.test(data$median_income[data$wasAtCMP == 1], data$median_income[data$wasAtCMP == 0])
```

```
##
##  Welch Two Sample t-test
##
## data:  data$median_income[data$wasAtCMP == 1] and data$median_income[data$wasAtCMP == 0]
## t = 4.8509, df = 470.61, p-value = 1.674e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   6722.967 15878.418
## sample estimates:
## mean of x mean of y
## 104581.79  93281.09
```

```
N <- 10000

diffGmea <- rep(NA, N)
actualdiff <- mean(data$median_income[data$wasAtCMP == 1]) - mean(data$median_income[data$was
AtCMP == 0])

for (i in 1:N) {
  genmea <- sample(data$wasAtCMP)  # default is replace = FALSE ----- taking a sample of the
labels here
  diffGmea[i] <- mean(data$median_income[genmea == 1]) -  mean(data$median_income[genmea ==
0])
}


hist(diffGmea, col = 3, main = "Permuted Sample Mean Diff in Income by CMP Status", xlab = "I
ncome", breaks = 50, xlim=c(-15000, 15000))
abline(v = actualdiff, col=4, lwd=3, cex = 0.8)
text(actualdiff-800, 500, paste("Actual Diff in Means =", round(actualdiff,2)), srt = 90)
```




```
#Two-sided p-value for difference of medians
mean(abs(diffGmea) >= abs(actualdiff))
```

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

The null hypothesis is that the difference in median income scores between teams that did and didn't qualify for the world championship is 0, and the alternative hypothesis is that the median income of those groups are not equal (the difference between them is not 0). We have an alpha level of .05 and a p-value, calcuated through permutation, of 0. Assuming the null hypothesis is true, there is a nearly 0% chance that a sample at least as extreme as this one would be observed. This result is exceedingly unlikely to have occured due to random chance alone – p exceeds alpha, so we must reject the null and conclude that the median incomes for teams which did and did not make it to world's are not equal.

Multiple Regression

Regression #1

My plan is to use best subsets regression and the Bayesian Information Criterion (BIC) to identify the team characteristics that best predict robot performance. I will then quantify this relationship with a linear model. Team performance is measured by each robot's calculated contributions to winning margin (CCWM) at the 2022 world championship.

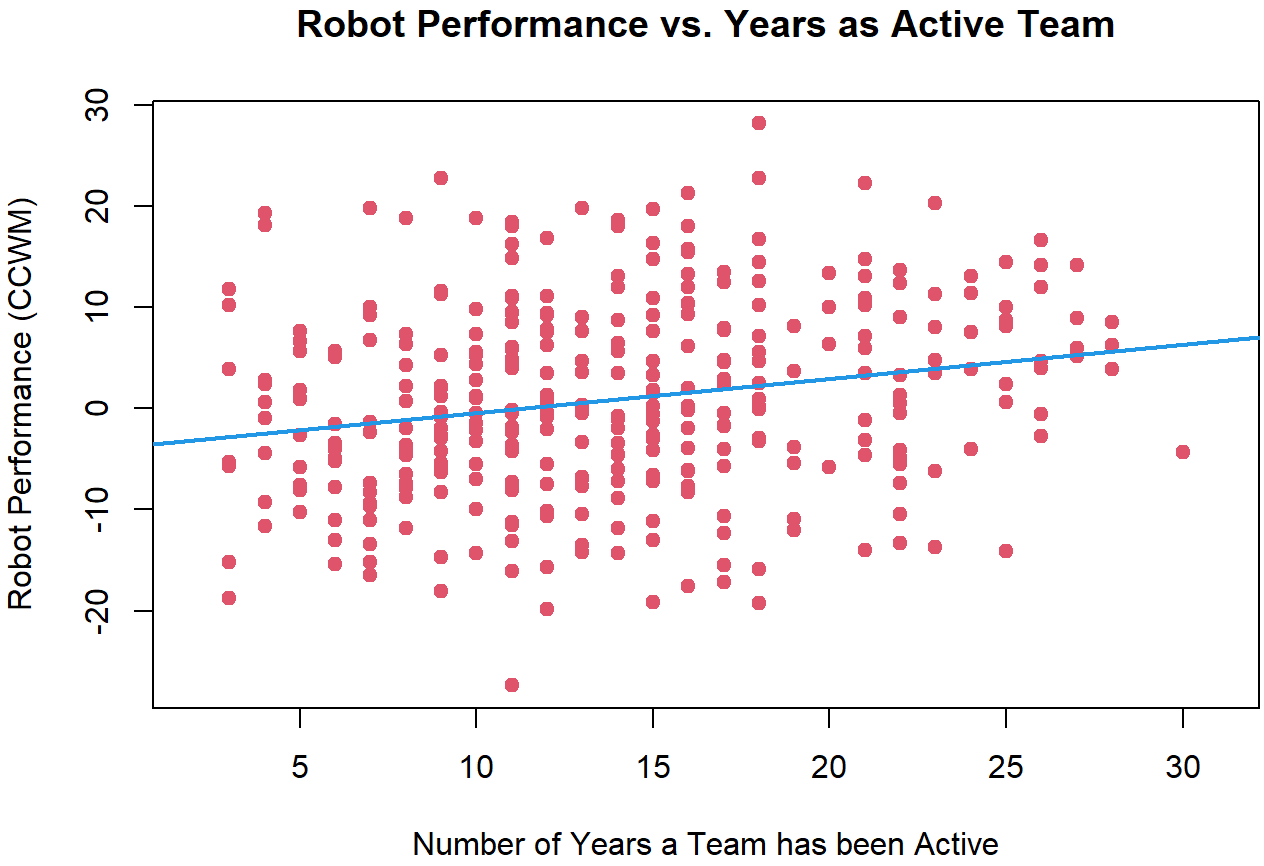
```
lm1 <- regsubsets(ccwm ~ log(median_income) + higher_ed_perc + pop_dens + numYears, data = da
ta)
lm1_sum <- summary(lm1)

modnum <- which(lm1_sum$bic == min(lm1_sum$bic)) # model with SMALLEST BIC
lm1_sum$which[modnum, ]
```

```
##        (Intercept) log(median_income)     higher_ed_perc           pop_dens
##               TRUE              FALSE              FALSE              FALSE
##           numYears
##               TRUE
```

```
lm2 <- lm(ccwm ~ numYears, data = data)
summary(lm2)
```

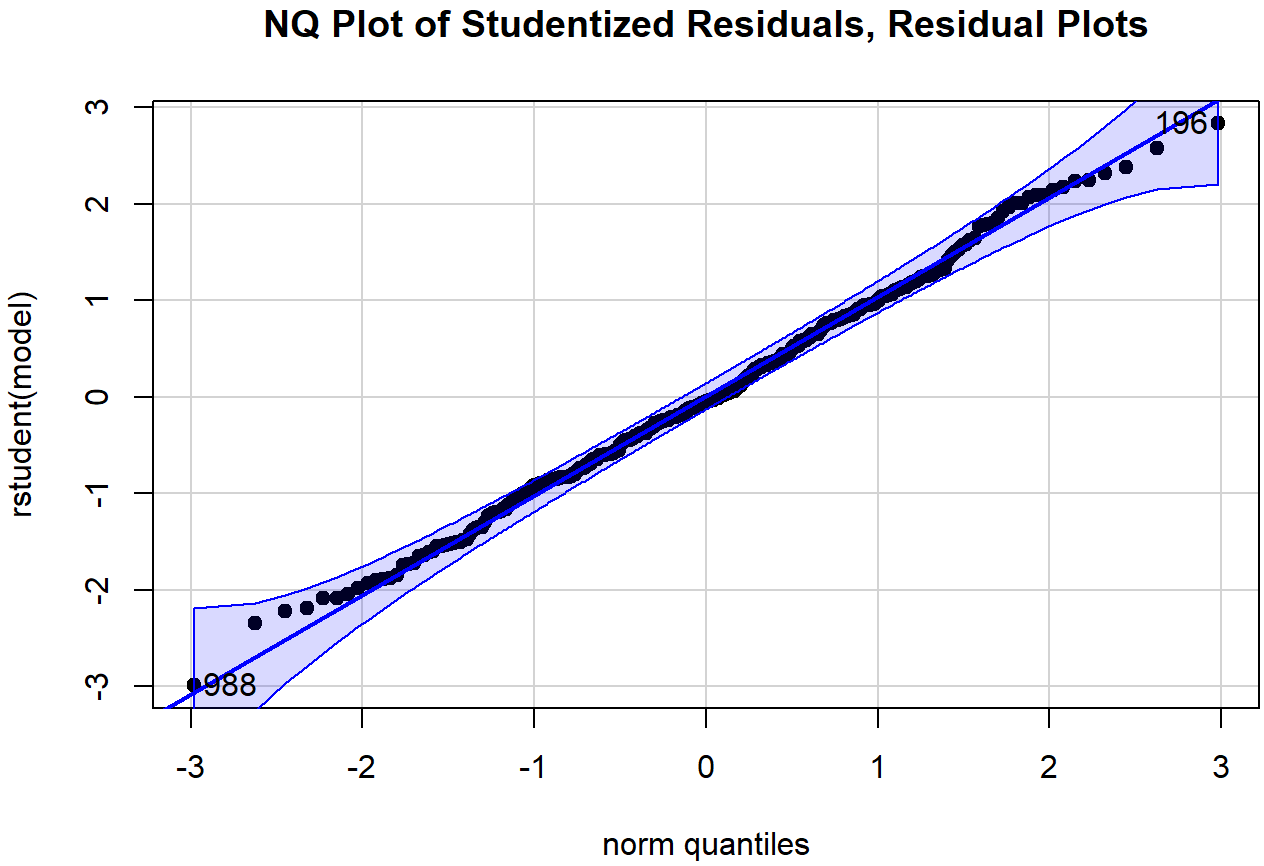
```
##
## Call:
## lm(formula = ccwm ~ numYears, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -27.2813  -6.3363  -0.4724   6.4861  25.8827
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3.7929     1.2395  -3.060  0.00239 **
## numYears      0.3369     0.0817   4.123 4.68e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.254 on 346 degrees of freedom
##   (1991 observations deleted due to missingness)
## Multiple R-squared:  0.04684,    Adjusted R-squared:  0.04408
## F-statistic:    17 on 1 and 346 DF,  p-value: 4.682e-05
```

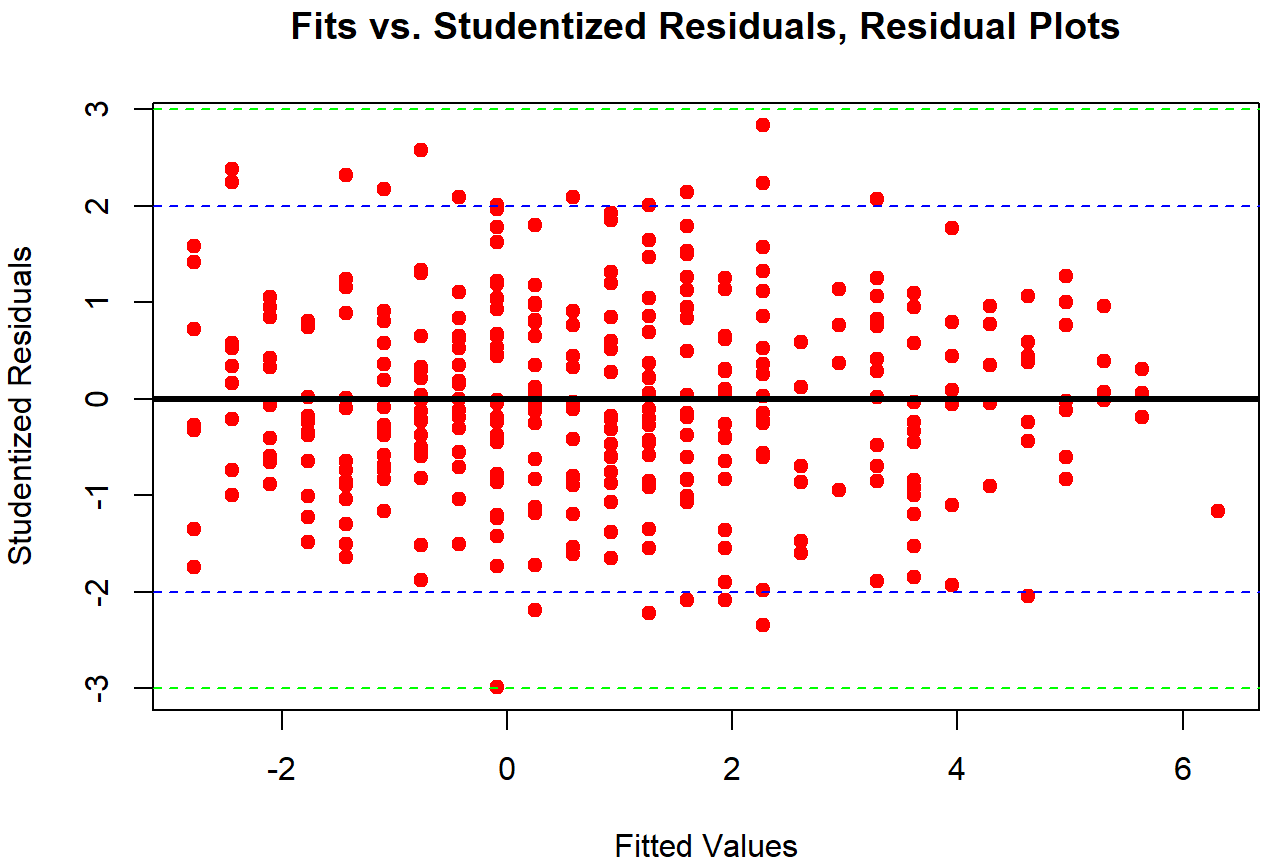
```
plot(data$ccwm ~ data$numYear, pch = 19, col = 2,
     xlab = "Number of Years a Team has been Active", ylab = "Robot Performance (CCWM)",
     main = "Robot Performance vs. Years as Active Team")
abline(lm2, lwd = 2, col = 4)
```



Out of the teams at the world championship, older robotics teams, as measured by the number of years the team has been active, tend to build robotics that perform better, as measured by the CCWM, than those built by newer teams. This relationship is statistically significant with a p-value of 4.68e-05, but not very explanatory: Variation in numYears only explains 4.68% of variation in CCWM, as indicated by the r-squared value. Its not suprising that the “best” model only used one explanatory variable, as the BIC gives a large penalty for using more parameters.

```
myResPlots2(lm2)
```

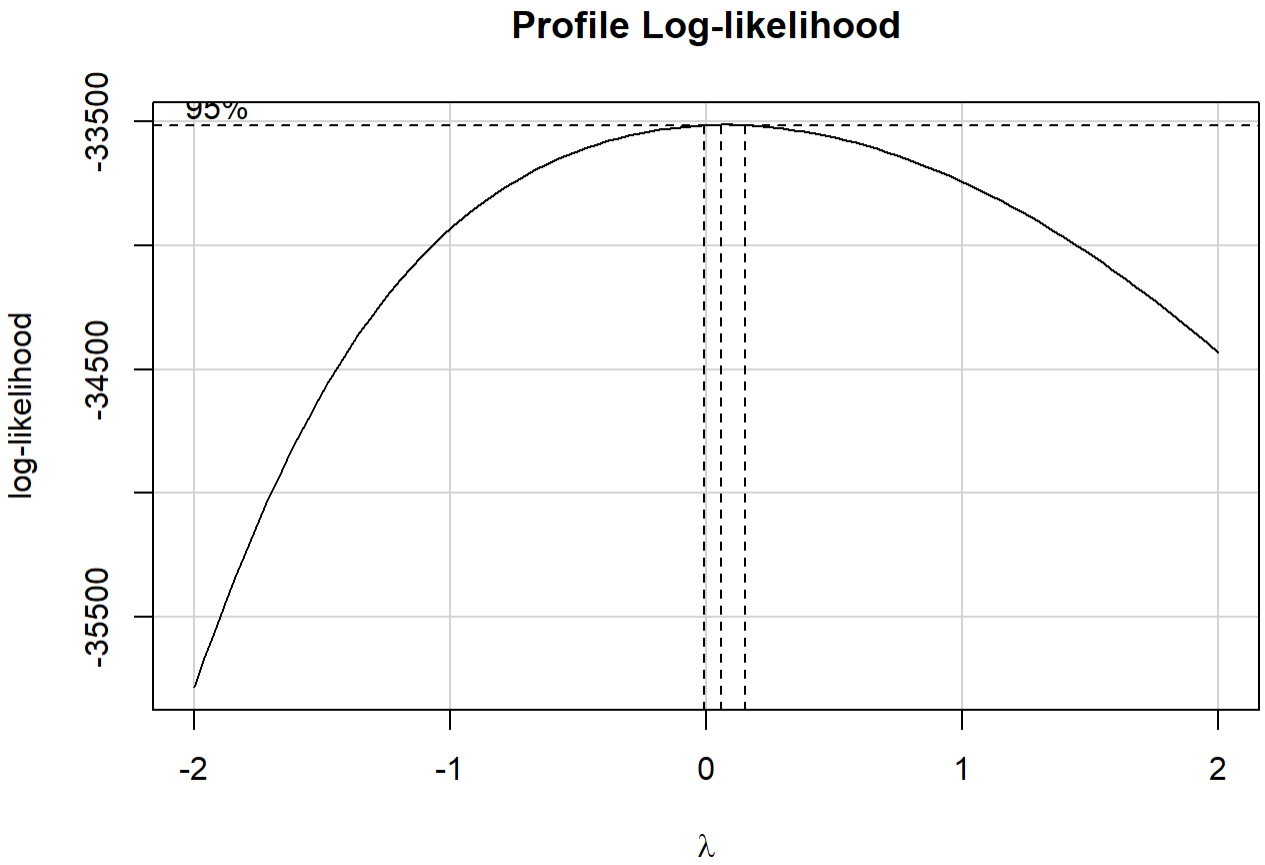

NQ Plot of Studentized Residuals, Residual Plots
rstudent(model)
norm quantiles
196
988
3
2
1
0
-1
-2
-3


Fits vs. Studentized Residuals, Residual Plots
Studentized Residuals
Fitted Values
3
2
1
0
-1
-2
-3
-2
0
2
4
6

Observations on the normal quantile plot are approximately linear, meaning the residuals are approximately normally distributed. Additionally, the plot of fits vs. residuals doesn't show any concerning hetereoskedasticity, outliers or overly influential points. This indicates that the model used is a good fit.

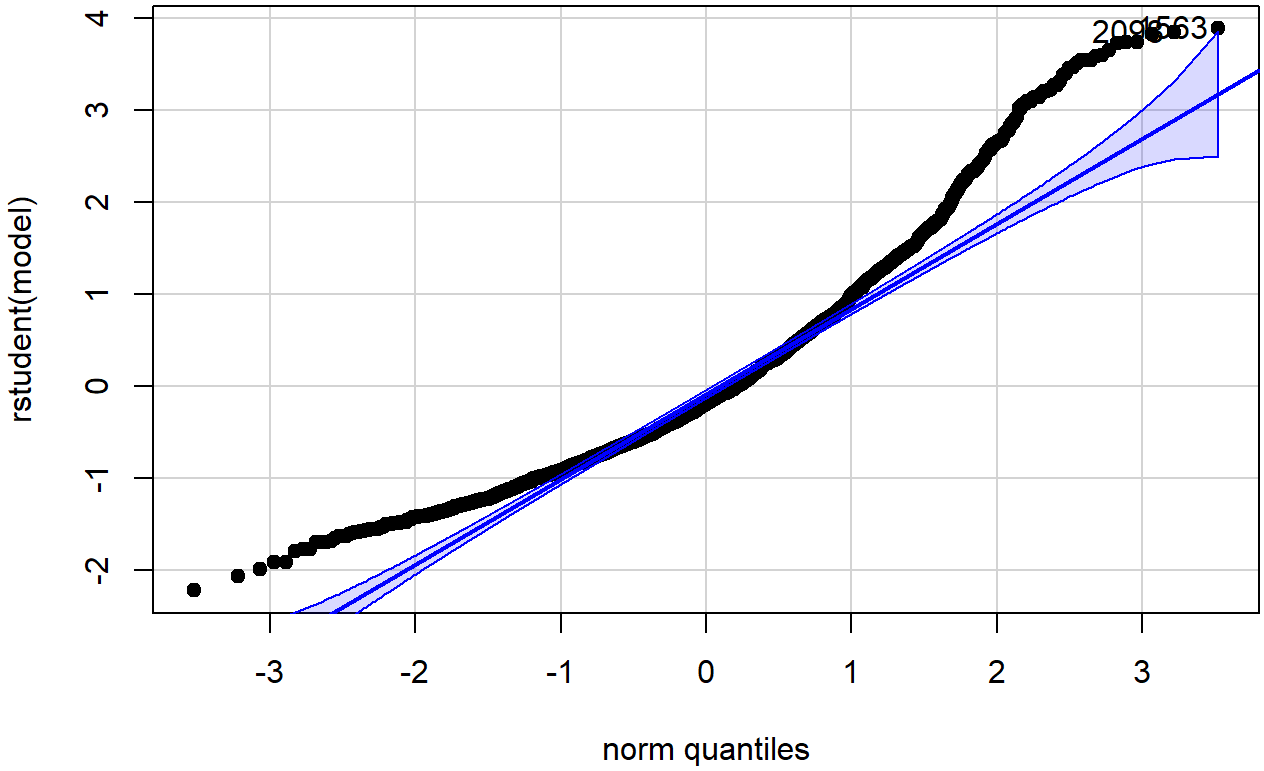
Regression #2

```
MultReg <- lm(data$median_income ~ data$numAwards)
boxCox(MultReg)
```

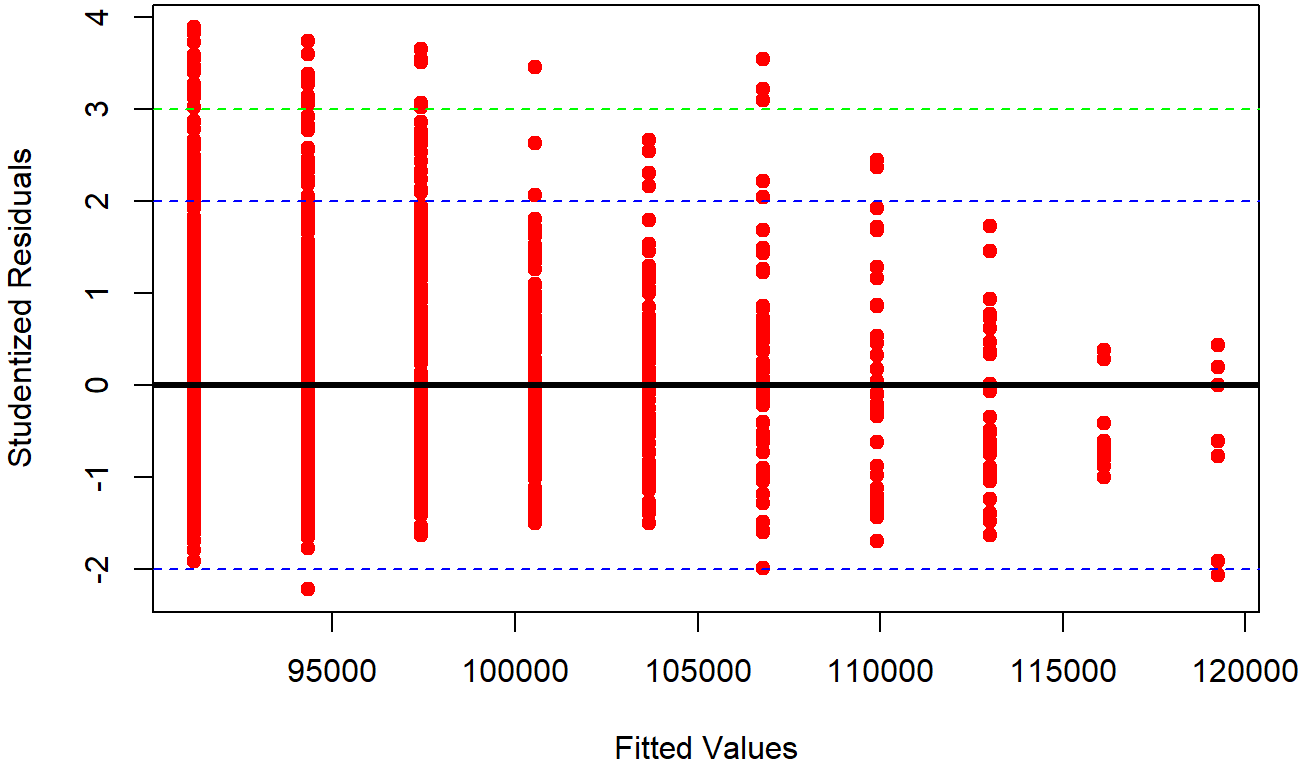



```
myResPlots2(MultReg)
```

NQ Plot of Studentized Residuals, Residual Plots



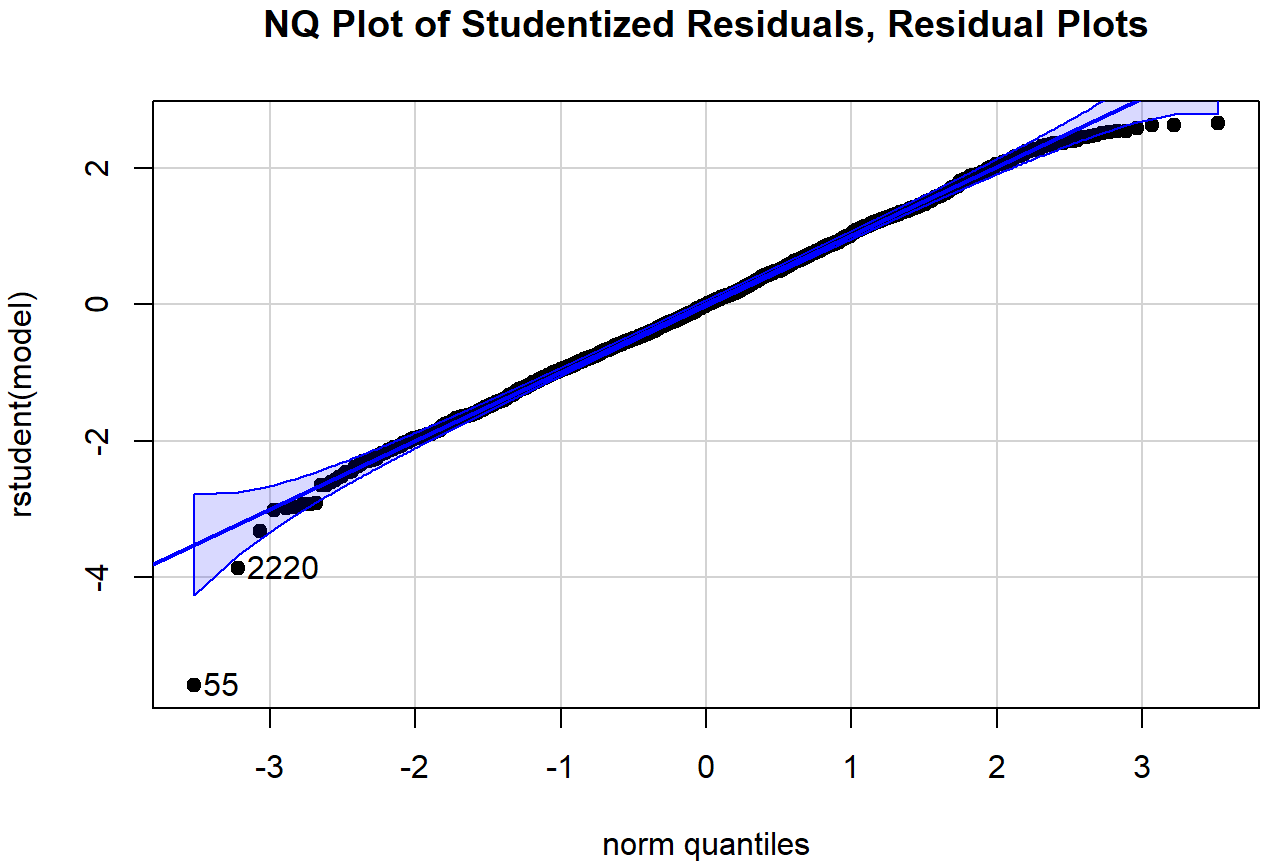
Fits vs. Studentized Residuals, Residual Plots

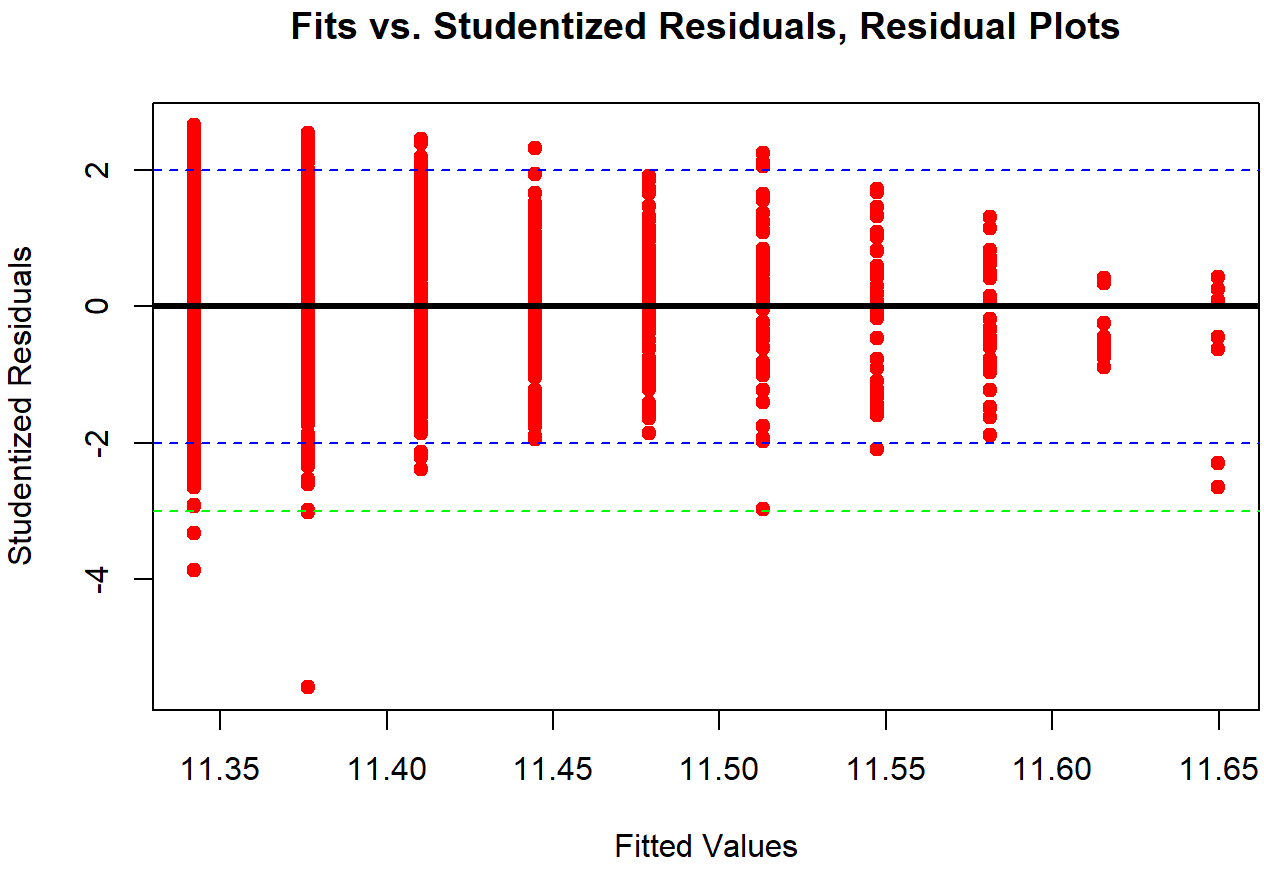


```
summary(MultReg)
```

```
##
## Call:
## lm(formula = data$median_income ~ data$numAwards)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -84482 -27257  -7337  20314 147343
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     91210.4      973.4   93.70  < 2e-16 ***
## data$numAwards   3115.0      469.9    6.63 4.17e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38070 on 2337 degrees of freedom
## Multiple R-squared:  0.01846,    Adjusted R-squared:  0.01804
## F-statistic: 43.95 on 1 and 2337 DF,  p-value: 4.169e-11
```

```
MultReg2 <- lm(log(data$median_income) ~ data$numAwards)
myResPlots2(MultReg2)
```


NQ Plot of Studentized Residuals, Residual Plots
rstudent(model)
norm quantiles
2220
55
2
0
-2
-4
-3
-2
-1
0
1
2
3


Fits vs. Studentized Residuals, Residual Plots
Studentized Residuals
Fitted Values
2
0
-2
-4
11.35
11.40
11.45
11.50
11.55
11.60
11.65

```
summary(MultReg2)
```

```
##
## Call:
## lm(formula = log(data$median_income) ~ data$numAwards)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.18180 -0.26181 -0.00507  0.27021  1.04022
##
## Coefficients:
##                Estimate Std. Error  t value Pr(>|t|)
## (Intercept)    11.34213    0.01005 1128.775  < 2e-16 ***
## data$numAwards  0.03419    0.00485    7.048 2.38e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.393 on 2337 degrees of freedom
## Multiple R-squared:  0.02081,    Adjusted R-squared:  0.02039
## F-statistic: 49.67 on 1 and 2337 DF,  p-value: 2.383e-12
```

Out of the robotics teams included in the dataset, teams that won more awards tend to be located in areas with a higher median income than teams that won fewer awards. The normal quantile plot was non-linear and the residual plot showed significant heteroskedasticity, so I ran a boxcox analysis, which suggested transforming median income using a natural log. After performing this transformation the normal quantile plot was linear, although there was still some heteroskedasticity in the residual plot. This relationship is significant, with a p-value of 2.38e-12, although of a small magnitude, with a multiple R-squared value of 0.02081.

Logistic Regression

```
logistic_regr_mod <- glm(data$wasAtCMP ~ data$higher_ed_perc + data$pop_dens + data$numYears,
family = binomial)
Anova(logistic_regr_mod, type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: data$wasAtCMP
##                     LR Chisq Df Pr(>Chisq)
## data$higher_ed_perc   19.789  1  8.646e-06 ***
## data$pop_dens          8.630  1   0.003306 **
## data$numYears         51.475  1  7.250e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(logistic_regr_mod)
```

```
##
## Call:
## glm(formula = data$wasAtCMP ~ data$higher_ed_perc + data$pop_dens +
##     data$numYears, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0223  -0.6091  -0.4978  -0.4034   2.6976
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -3.189e+00  2.053e-01 -15.536  < 2e-16 ***
## data$higher_ed_perc  1.527e-02  3.444e-03   4.433 9.27e-06 ***
## data$pop_dens       -3.795e-05  1.551e-05  -2.447   0.0144 *
## data$numYears        6.610e-02  9.164e-03   7.213 5.48e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1998.6  on 2338  degrees of freedom
## Residual deviance: 1909.3  on 2335  degrees of freedom
## AIC: 1917.3
##
## Number of Fisher Scoring iterations: 5
```

Using a logistic regression model is an appropriate model for this instance as we are seeking to project the odds that a given team made it to the world championship, a dichotomous outcome. Using a link function as the logit model does to connect the underlying binary probability distribution to a continuous one is what allows us to model the outcome likelihood better than a linear probability model since it will not include probabilities which are impossible, and because the nonlinear form does a better job at minimizing residuals in the middle range of probabilities.

Looking at the modeling results, one finds that a one unit increase in percent of adults with a college degree in the zip code or number of years the team has been around increase the log odds of a team making it to world's by .015 and .066 respectively indicating that both are positive relationships which are significant to 99.99% confidence. There is also a negative relationship between the population density of a school's zip code and the log odds of being at world's, for which the log odds decrease by .000038 for every unit increase in population density, still significant but only to 95% confidence.

Conclusions and Summary

In conclusion, socio-economic indicators, such as median household income, educational attainment, and population-density in a team's home zipcode are meaningfully associated with robot team characteristics like team age and performance at competition. To highlight a couple of these relationships: Higher median income is associated with better team performance. Higher educational attainment is associated with older teams, and a team being older is associated with it performing better. With that said, the explanatory power of these predictor variables is low. This makes sense, because many variables that weren't accounted for contribute to team success, like grants, particularly engaged students, and luck. The positive direction of the association isn't

surprising to me either. Greater income and higher educational attainment in a zip code may offer the team better access to monetary and mentorship resources. By better understanding the factors which contribute to a team's success, more targeted equity programs can be planned that better address team needs and counter inequality in access to the quality STEM education that a robotics team can provide.