Analysis of Attending Physicians and Proportion of Black Mothers

Sheila Braun

August 6, 2018

Contents

Data Preparation

This data set relates to doctors and births in the state of a state in the US. A series of questions from a researcher guides the analysis below.

The first step is to load and check the data, then fix anything that needs fixing. As usual, I will name the data by its structure, in this case *dt* for "data table" or to be more precise "data tibble" (thanks to Hadley Wickham for that data structure).

This document contains a call to the data file provided by the researcher and all the code necessary to reproduce any findings. It also functions as a report. The document, then, is code, analysis, and report in one. To reproduce results, simply "knit" the report file in RStudio or similar, taking care to put the data file in the same folder and with the same name as in this document.

[Please note that you can ignore the various warnings throughout. They have been checked and found harmless.]

Load the File

```
library(dplyr)
library(magrittr)
fname <- "exercise.csv"
dt <- as_tibble(read.csv(fname))
```

Check the Data Structure

```
str(dt)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    52027 obs. of  13 variables:
##  $ PAF   : int  400 400 400 400 400 400 400 400 400 400 ...
##  $ SEX   : Factor w/ 3 levels "F","M","U": 2 2 1 1 2 2 2 2 2 1 ...
##  $ RACE  : Factor w/ 11 levels "","1","7","A",..: 11 10 11 10 11 11 11 11 10 11 ...
##  $ ID    : Factor w/ 39289 levels "","{{{F442501",..: 17386 19519 29189 18177 4141 33393 16624 5253 1
##  $ AGE   : int  56 3 68 20 57 41 65 66 0 68 ...
##  $ PZIP  : Factor w/ 992 levels "","00000","00757",..: 989 470 446 487 453 469 488 486 602 432 ...
##  $ DX1   : Factor w/ 2881 levels "","0030","0039",..: 275 16 933 1294 1929 2784 1051 2244 39 2823 ..
##  $ DX2   : Factor w/ 3282 levels "","0030","0048",..: 530 447 952 1374 2040 3024 1943 1022 447 1567
##  $ DX3   : Factor w/ 3049 levels "","0039","0071",..: 217 1 843 1 1 1 64 843 9 40 ...
##  $ PR1   : int  38 NA 3601 470 8051 8674 8611 5491 8607 NA ...
##  $ PR2   : int  9928 NA 3722 5421 309 863 863 NA NA NA ...
##  $ PR3   : int  331 NA 8853 544 NA 8669 8623 NA NA NA ...
##  $ ATT_ID: Factor w/ 1820 levels "","0550476","0S007185L",..: 1169 72 1381 932 1233 1386 81 1105 103
```

Drop Irrelevant Variables and Format Variable Names

Only a few variables in the list above are necessary for this analysis. We have two copies of the data set: one in the file on the drive, which must not be changed, and the other in memory. It is best to keep memory as clear as possible, so we will drop variables from memory that are unrelated to the analysis. Also, for ease of coding, we will lowercase all the variable names and then uppercase them for the final dataset export.

```
names(dt) <- tolower(names(dt)) # note to self: remember to reset this at the end
dt <- dt %>%
        dplyr::select(-dx1,
               -dx2,
               -dx3,
               -pzip,
               -paf,
```

```
                -age,
                -sex) # the variables *pr1*, *pr2* and *pr3* related to births are gender proxies
```

Missing Values

Check to see if there are any missing values in the data set.

```
any(is.na(dt[]))
```

```
## [1] TRUE
```

This value indicates that there is at least one missing value. It's a good idea to understand any missing values so we know whether they affect the planned analysis. First find out how many there are:

```
sum(is.na(dt[]))
```

```
## [1] 92888
```

92888 is a lot of missing values. Now check where they are in the data table:

```
sapply(dt, function(x) sum(is.na(x)))
```

```
##   race     id    pr1    pr2    pr3 att_id
##      0      0  19840  32929  40119      0
```

The procedure variables *pr1*, *pr2*, and *pr3* have missing values.

Filtering for specific procedures related to giving birth, which is in the next step, will eliminate any cases missing values for all three procedure columns.

Select Birth-Related Cases

Next use indexing to keep only the procedure codes related to delivery of babies. The logic is to include in dt all those cases for which *pr1* or *pr2* or *pr3* match a number in the list of delivery codes. This logic excludes anyone who comes to the hospital and leaves again without having a delivery-related procedure during that visit.

```
delivery_codes <- c(720, 721, 724, 726, 728, 729,
                    731, 733, 736, 738,
                    740, 741, 742, 744)
dt <- dt[dt$pr1 %in% delivery_codes |
                 dt$pr2 %in% delivery_codes |
                 dt$pr3 %in% delivery_codes, ]
```

```
sapply(dt, function(x) sum(is.na(x)))
```

```
##   race     id    pr1    pr2    pr3 att_id
##      0      0      0    609   1388      0
```

The above table shows that there are no longer any cases that are missing *pr1*. Each of the remaining 2700 visits includes at least one procedure related to having a baby.

1. Number of doctors in the state delivering babies

Count the attending doctors.

```
library(data.table)
num_atts <- uniqueN(dt$att_id) # Benchmark variable for the tests below.
```

The number of doctors doing deliveries in this data set is 176.

2. Average number of deliveries and c-sections per doctor

We calculate the number of deliveries and c-sections for each doctor and then the average number of deliveries and c-sections across all doctors. These numbers are stored in the dataset as new variables *num_births* and *cx*.

Average Births per Doctor

```
dt <- dt %>%
        group_by(att_id) %>%      #group the data by physician
        mutate(num_births = n())  #count births per dr, add as a new column
avg_deliveries <- round(mean(dt$num_births, na.rm = TRUE), 2) #calculate
```

The average number of deliveries per doctor is 24.64.

Average C-Sections per Doctor

Flag all c-sections and store the flags in variable *cx*.

```
c_section_codes <- c(740, 741, 742, 744) #provided by researcher
dt$cx <- apply(dt[, 3:5], 1, function(row) any(row %in% c_section_codes) )
dt <- dt %>%
        group_by(att_id) %>%   #group by attending physician
        mutate(cx_total = sum(cx))  #add newe column of total csections
avg_cx <- round(mean(dt$cx_total, na.rm = TRUE), 2)
```

The average number of c-sections per doctor is 8.41.

3. An easy-to-understand description of the distribution of deliveries per doctor

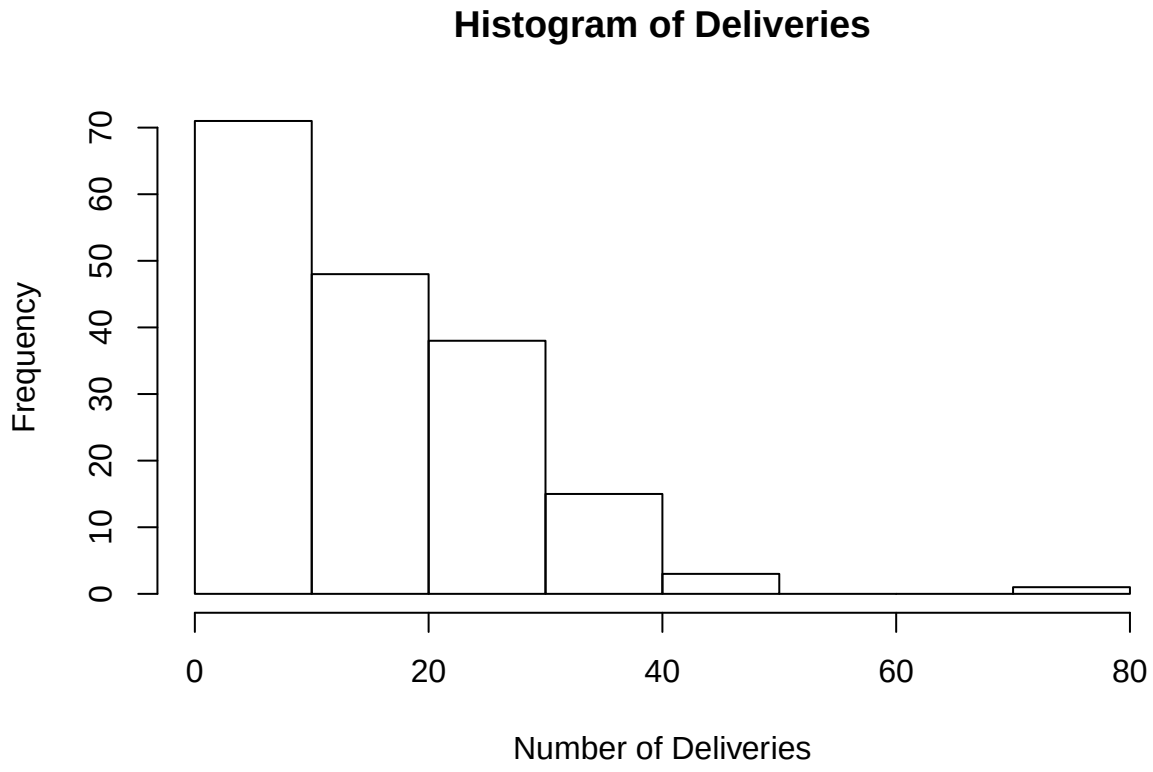
Is this a skewed distribution? What are other important characteristics of this distribution?

Examine Distributions

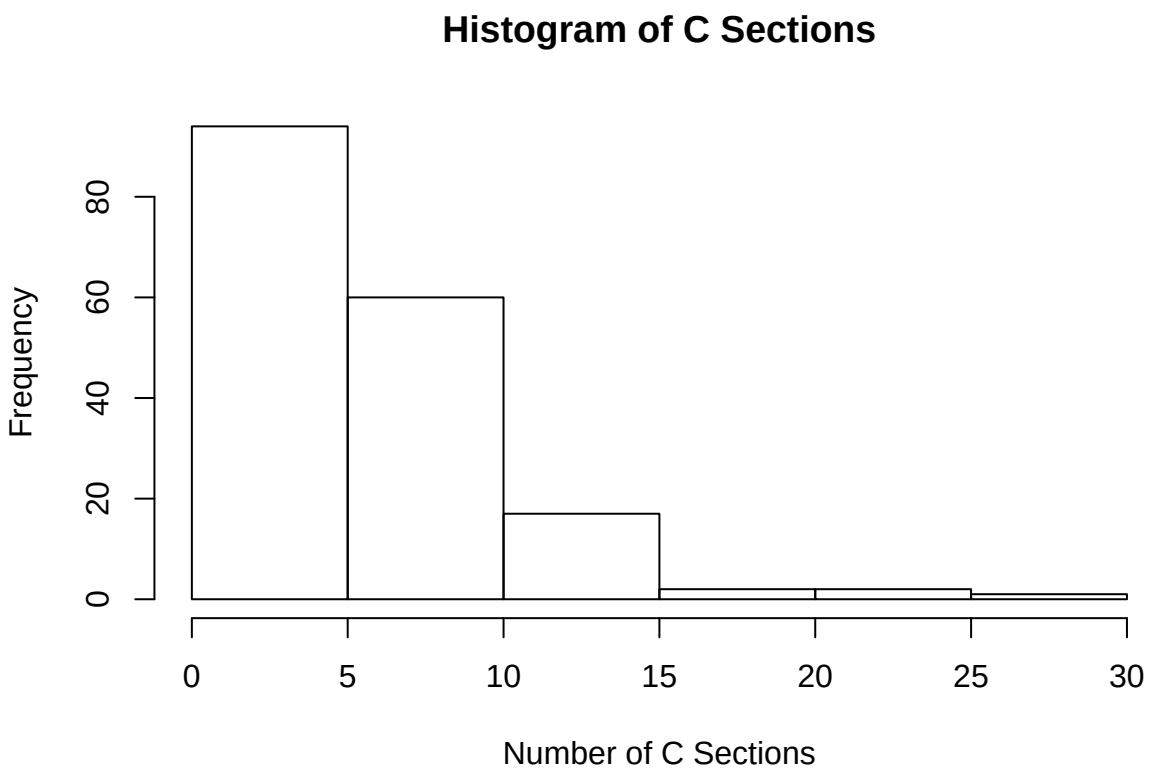
First, we'll look at histograms, and then at density plots. Histograms tell the frequency for each count of births; density plots tell the density of each count of births, using curved lines. The density plot shows data at a more detailed level.

```
library(ggplot2)
library(ggjoy)
library(ggridges)
library(scales)
```

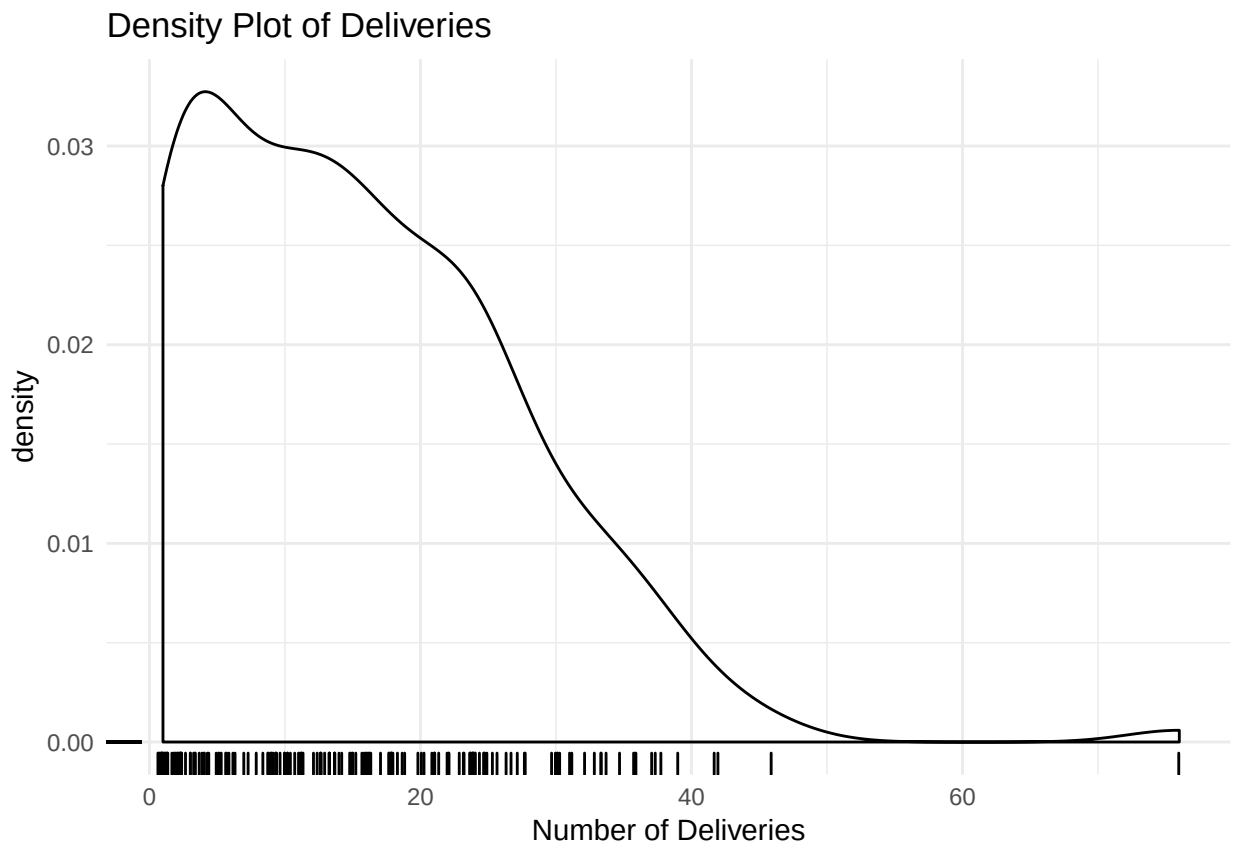
```r
library(lubridate)
library(ggthemes)
library(MASS)
library(lattice)

#first get data about doctors rather than individual cases

dtdr <- dt %>%
        dplyr::select(att_id,
                      num_births,
                      cx_total) %>%
        unique()

#all deliveries
hist(dtdr$num_births,
     xlab = "Number of Deliveries",
     main = "Histogram of Deliveries")
```




```r
#c sections
hist(dtdr$cx_total,
     xlab = "Number of C Sections",
     main = "Histogram of C Sections")
```




```
#all deliveries
ggplot(dtdr,
       aes(num_births,
           fill = dtdr$num_births)) +
        geom_density(aes(fill = dtdr$num_births),
                     alpha = 3/4,
                     na.rm = FALSE) +
        guides(fill = FALSE, color = FALSE) +
        labs(title = "Density Plot of Deliveries",
             x = "Number of Deliveries") +
        geom_rug(aes(x = dtdr$num_births, y = 0),
                 position = position_jitter(height = 0)) +
        theme_minimal()
```



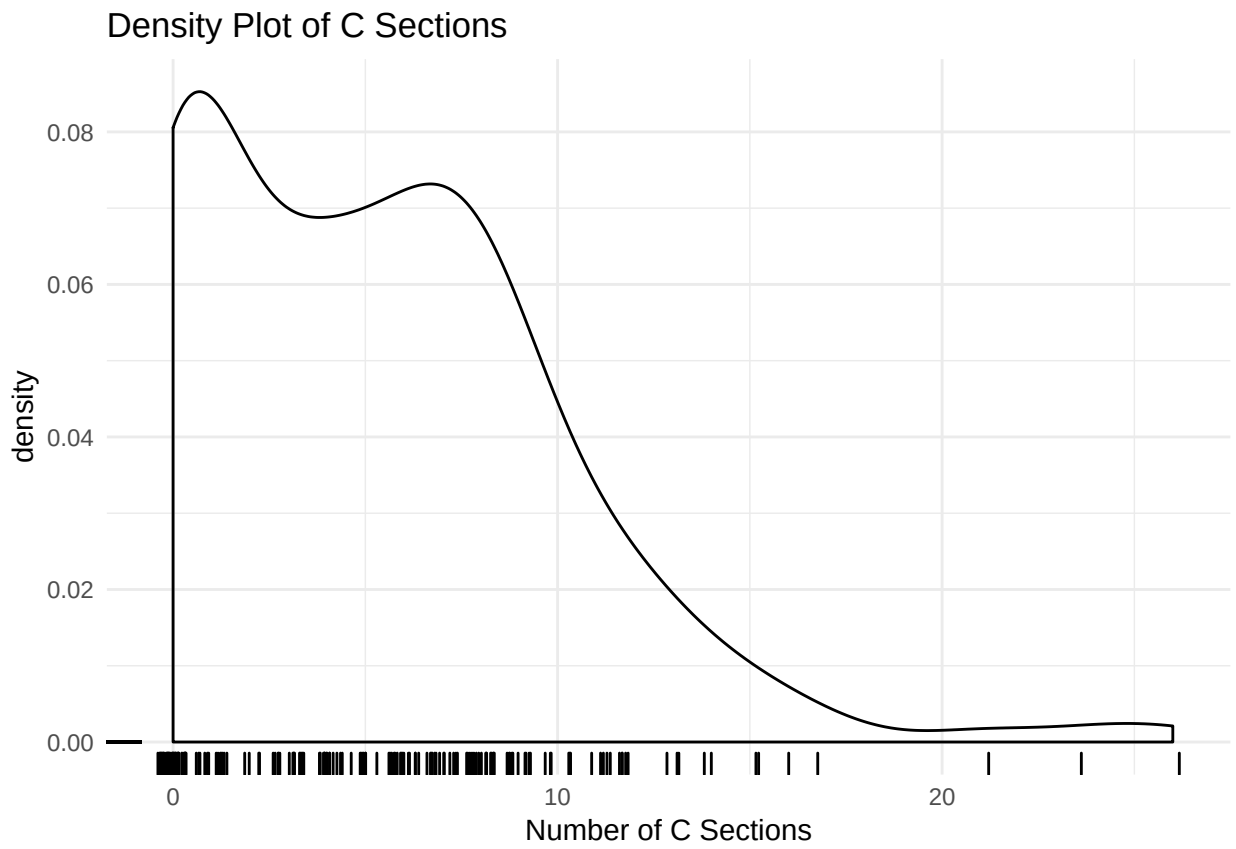
Interpret Plots

Density plots are slightly harder to interpret than the histograms are, but provide a better sense of the shape of the data.

The data set for all deliveries has the following qualities:

- Data are skewed to the right, which means that the number of highly experienced doctors is more spread out and lower than than the number of relatively inexperienced doctors.
- Almost all doctors have done fewer than 40 deliveries
- A large proportion have done fewer than 25 deliveries.

```
#C-sections
ggplot(dtdr,
       aes(cx_total,
           fill = dtdr$cx_total)) +
        geom_density(aes(fill = dtdr$cx_total),
                     alpha = 3/4,
                     na.rm = FALSE) +
        guides(fill = FALSE, color = FALSE) +
        labs(title = "Density Plot of C Sections",
             x = "Number of C Sections") +
        geom_rug(aes(x = dtdr$cx_total, y = 0),
                 position = position_jitter(height = 0)) +
        theme_minimal()
```



The data set for all c-sections has the following qualities:

- Data are skewed to the right.
- The distribution of births and the distribution of births that are c-sections are similar.
- The majority of doctors who have done c-sections in a state in the US have done fewer than 10 of them.
- Almost every doctor has done fewer than 20 c-sections.

Note that the first histogram and density plot include the second, in that the first plots show all births *including* c-sections and the second is *only* c-sections. They are not independent populations. This analysis makes no assumption that they are, but it is good to keep this fact in mind for any later analyses.

It is also interesting to note the "empty space" between the number of deliveries among low to moderately experienced doctors and highly experienced doctors. This phenomenon bears further investigation because it raises (but does not answer) questions like whether there exists a lack of training opportunties for moderately experienced doctors to become highly experienced or whether, at this time, fewer doctors are choosing to become highly experienced at delivering babies.

4. The percentage of black mothers delivering babies (overall, and then per doctor)

```
library(tidyr)
# percentage of black mothers
n <- nrow(dt)
```

```
nblack <- nrow(dt[dt$race == "B", ])
proportion_black <- nblack / n
perc_black <- percent(proportion_black)
```

Overall Percentage of Black Mothers

Only 3.81% of the deliveries were with black mothers.

Proportion of Black Mothers: Per Doctor

```
library(Hmisc)
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
dt$mom_is_black <- apply(dt[, 1], 1, function(row) any(row %in% c("B")) )

length(dt$mom_is_black[dt$mom_is_black == TRUE]) - nblack #check it.
```

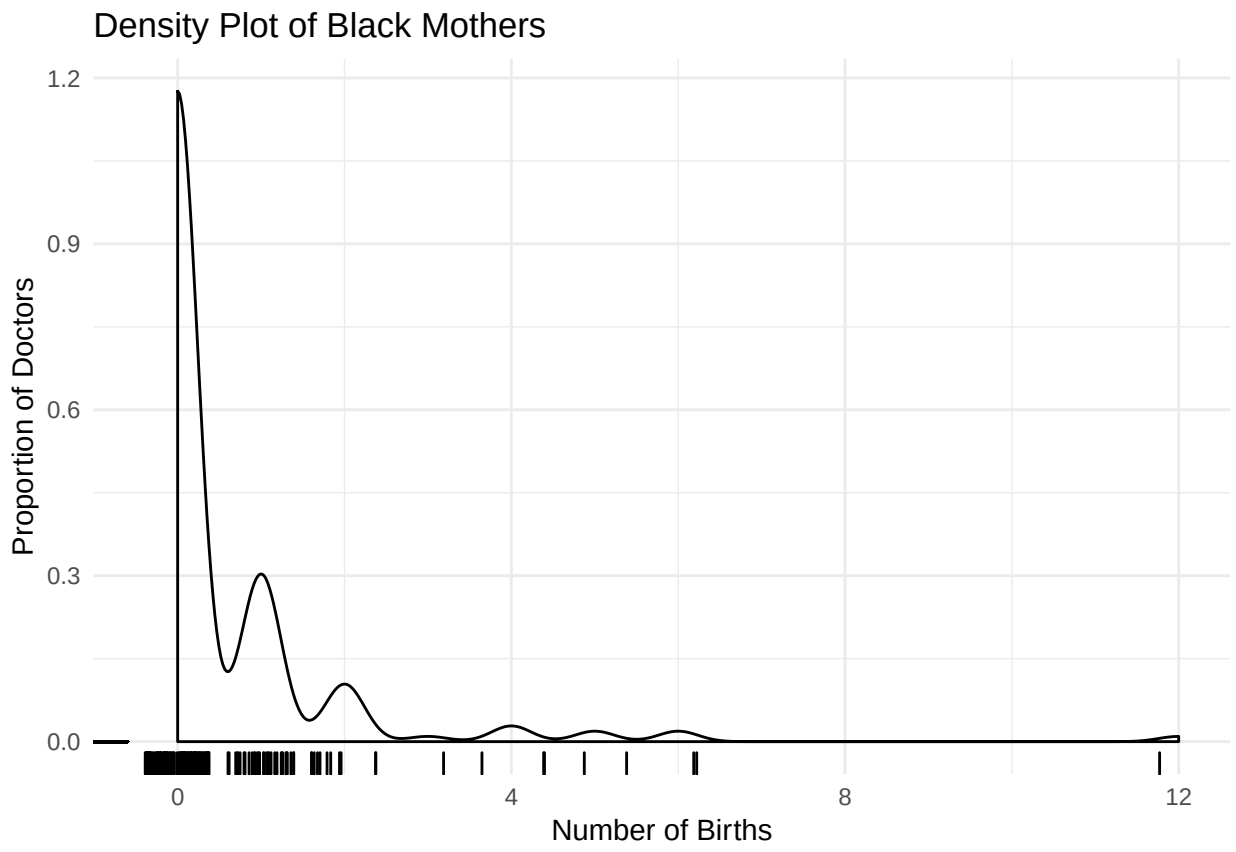
```
## [1] 0
```

```
dt <- dt %>%
        group_by(att_id) %>%
        mutate(black_total = sum(mom_is_black))
```

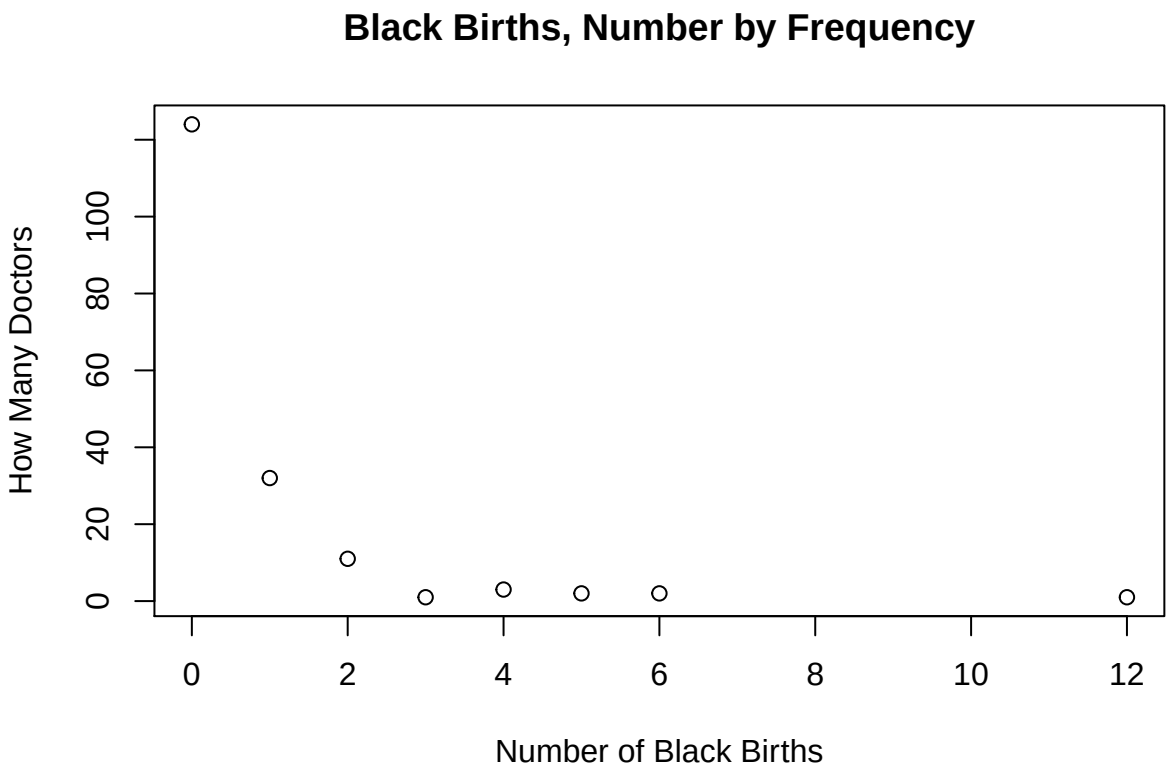
From here on, the data set is reduced to the number of doctors and summary statistics about them.

```
#do a little cleanup

dt <- dt %>% dplyr::select(att_id,
                           num_births,
                           cx_total,
                           black_total)
#select by physician
dt <- unique(dt)
dt <- dt %>% mutate(perc_black = percent(black_total / num_births))

#density plot of black births

ggplot(dt,
       aes(black_total,
           fill = dt$black_total)) +
        geom_density(aes(fill = dt$black_total),
```

```
                alpha = 3/4,
                na.rm = FALSE) +
    guides(fill = FALSE, color = FALSE) +
    labs(title = "Density Plot of Black Mothers",
         y = "Proportion of Doctors",
         x = "Number of Births") +
    geom_rug(aes(x = dt$black_total, y = 0),
             position = position_jitter(height = 0)) +
    theme_minimal()
```




```
description <- Hmisc::describe(dt)
bt <- description$black_total
plot(bt$values$value,
     bt$values$frequency,
     main = "Black Births, Number by Frequency",
     xlab = "Number of Black Births",
     ylab = "How Many Doctors")
```



This plot identifies an outlying doctor who has attended 12 black mothers. Most doctors have attended none, and all the other doctors have attended six or fewer. The outlier explains the gap we noted earlier. However, it remains in the data set because it is not an error. Being unique isn't in itself a good reason to eliminate a data point. First we will try to do the appropriate analysis that includes it. Although this point should be measured, it should not exert undue leverage on any calculations. An influential point is an outlier that greatly affects the slope of the regression line. One way to test the influence of an outlier is to compute the regression equation with and without the outlier. This is done below.

5. Are the patients of high volume doctors more likely to be black than the patients of low volume doctors?

Setting Up the Analysis

Numerically, the question at hand is whether having a high volume of patients predicts that a doctor will have more black patients. Therefore

```
Dependent variable = Percentage (or proportion) of mothers who are black
Predictor variable = Low-volume Doctor vs. High Volume Doctor
```

First define high-volume doctors as any who attended more than the mean number of births. The resulting factor variable is *hi*.

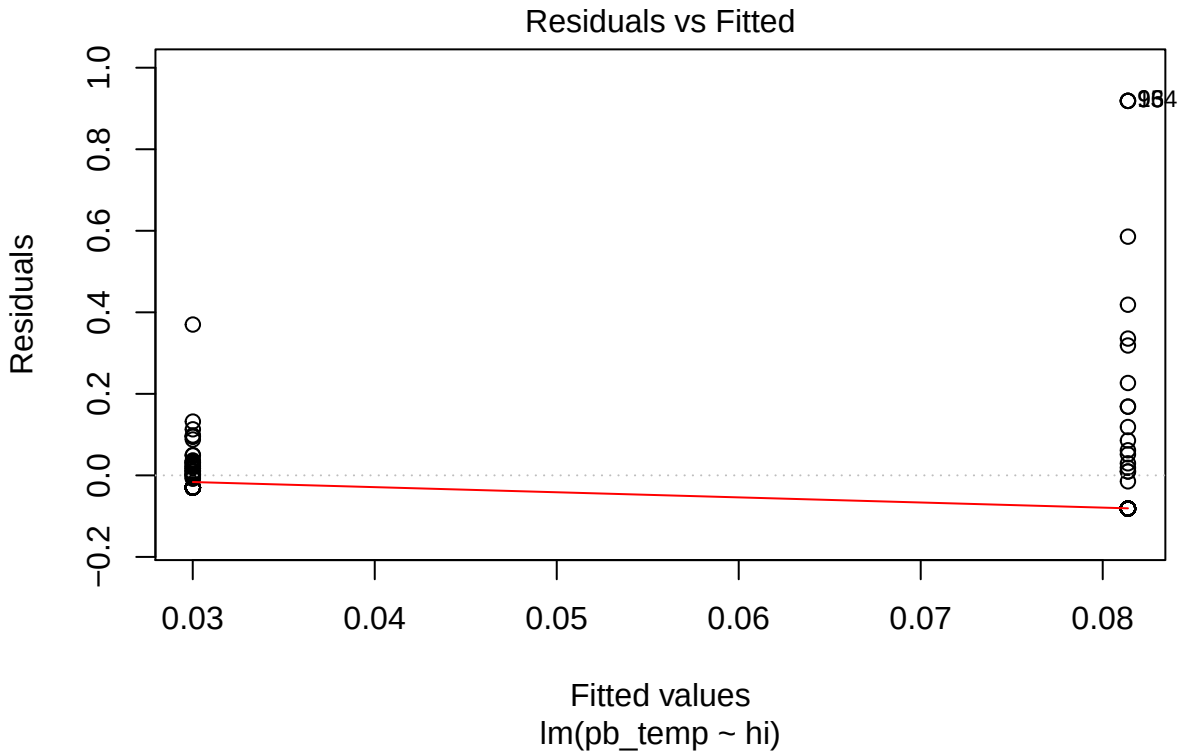
```
dt$hi <- apply(dt[, 2], 1, function(row) any(row > mean(dt$num_births)) )
dt$hi = dt$hi + 1 - 1 #convert "hi" from logical to numeric
dt$hi <- factor(dt$hi,
```

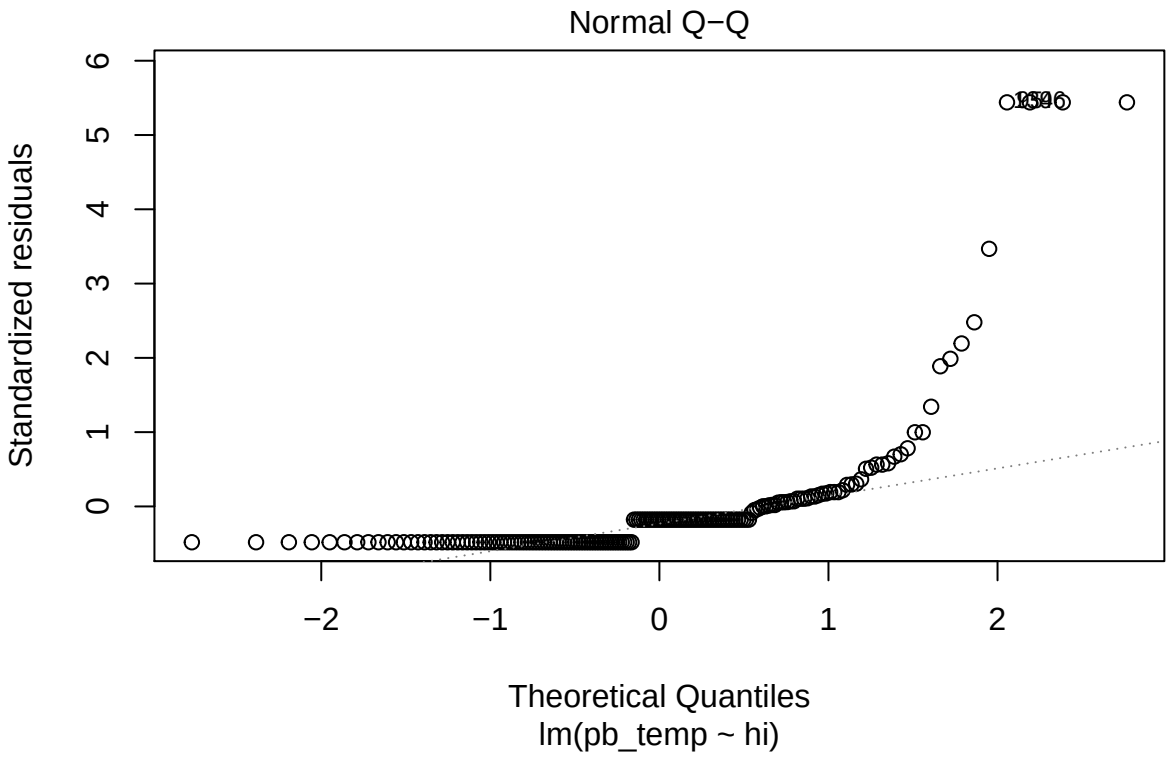
```
                levels = c(0, 1),
                labels = c("Low", "High"))
```

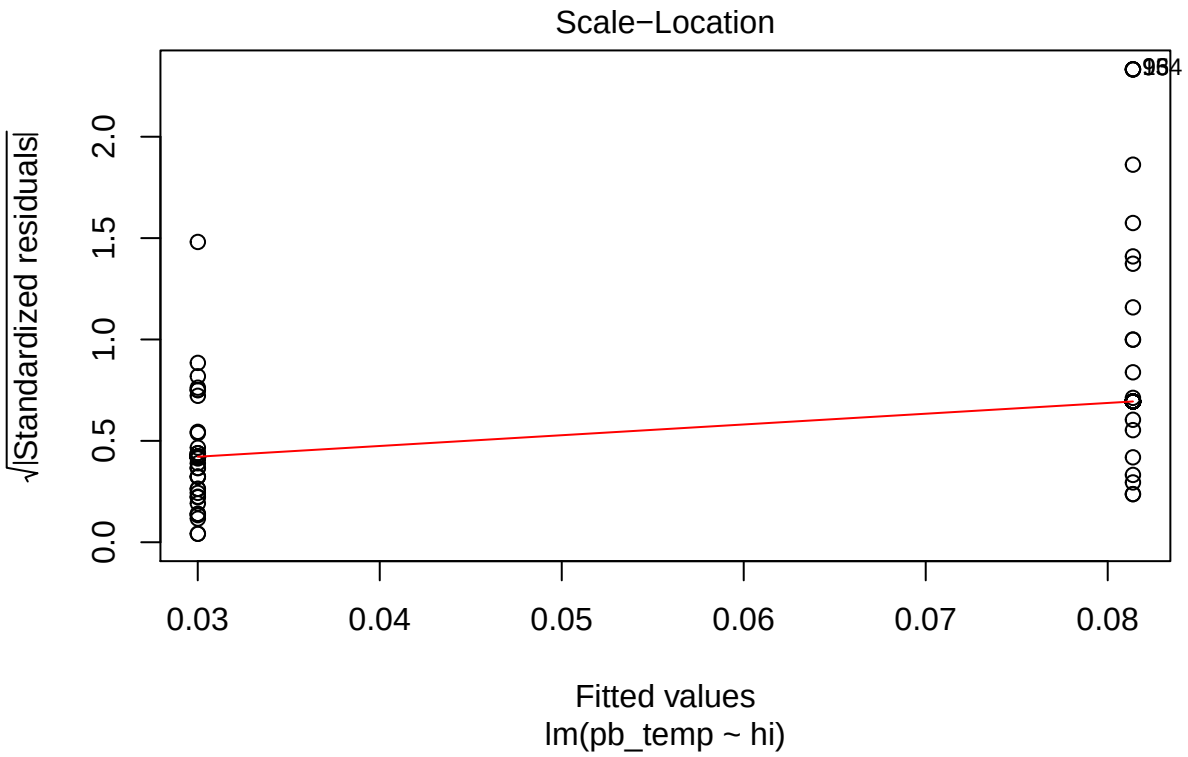
Analysis

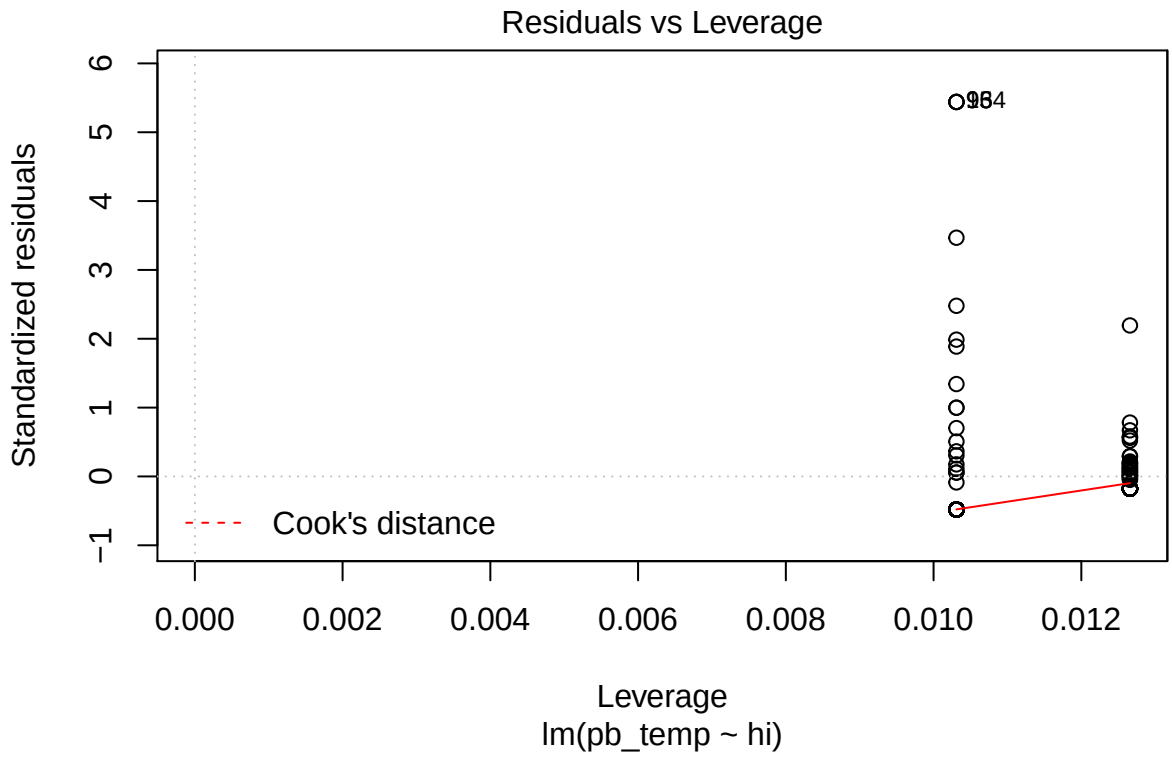
Now see whether and to what extent being a high-volume doctor predicts a high proportion of black mothers.

```
library(lmtest)
#create a non-character variable with the percentages:
dt$pb_temp <- as.numeric(sub("%", "", dt$perc_black, fixed = TRUE))/100

# linear regression where dependent is pb_temp
# and independent = high vs. low volume:
fit <- lm(pb_temp ~ hi, dt)
plot(fit)
```




Normal Q−Q
Standardized residuals
0
1
2
3
4
5
6
−2
−1
0
1
2
Theoretical Quantiles
lm(pb_temp ~ hi)

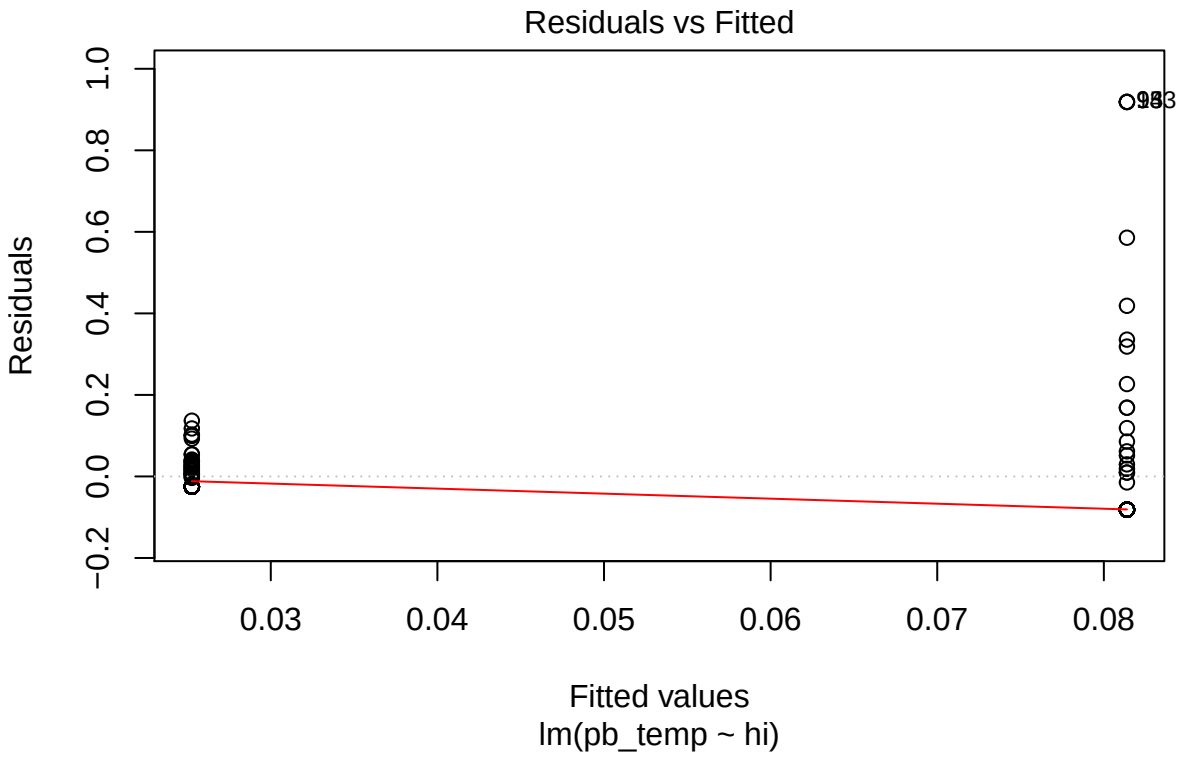

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
0.03
0.04
0.05
0.06
0.07
0.08
Fitted values
lm(pb_temp ~ hi)

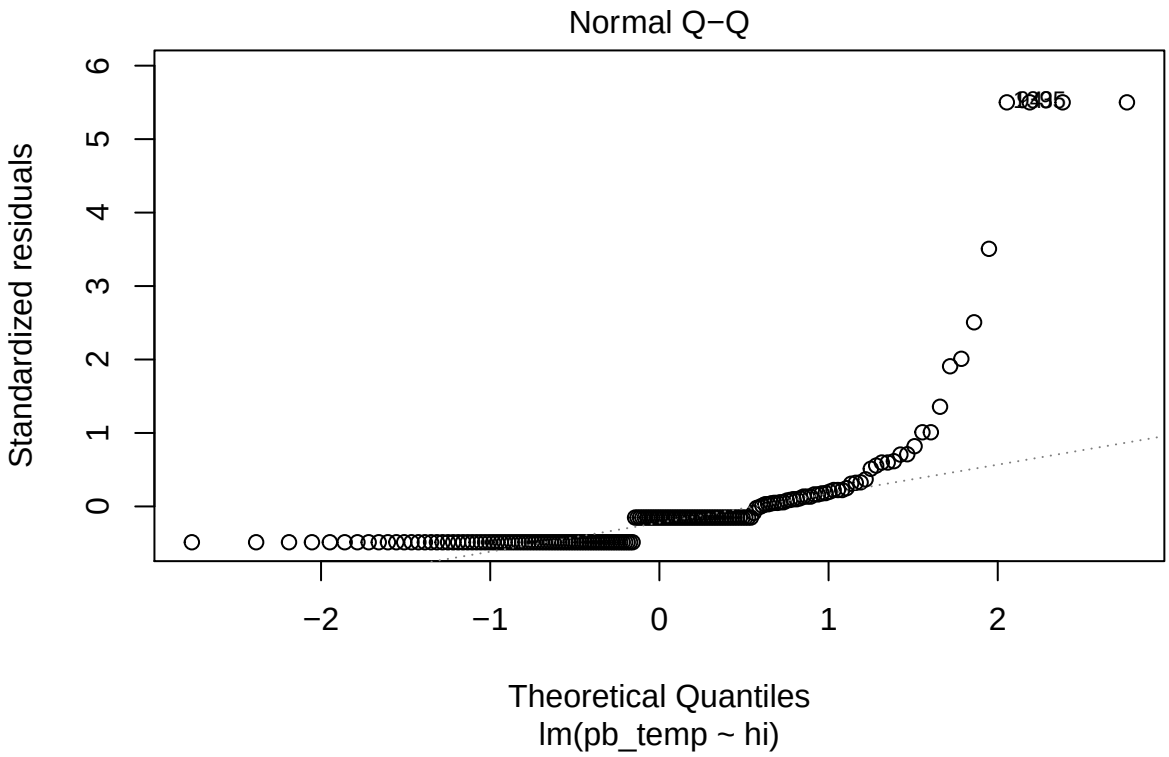


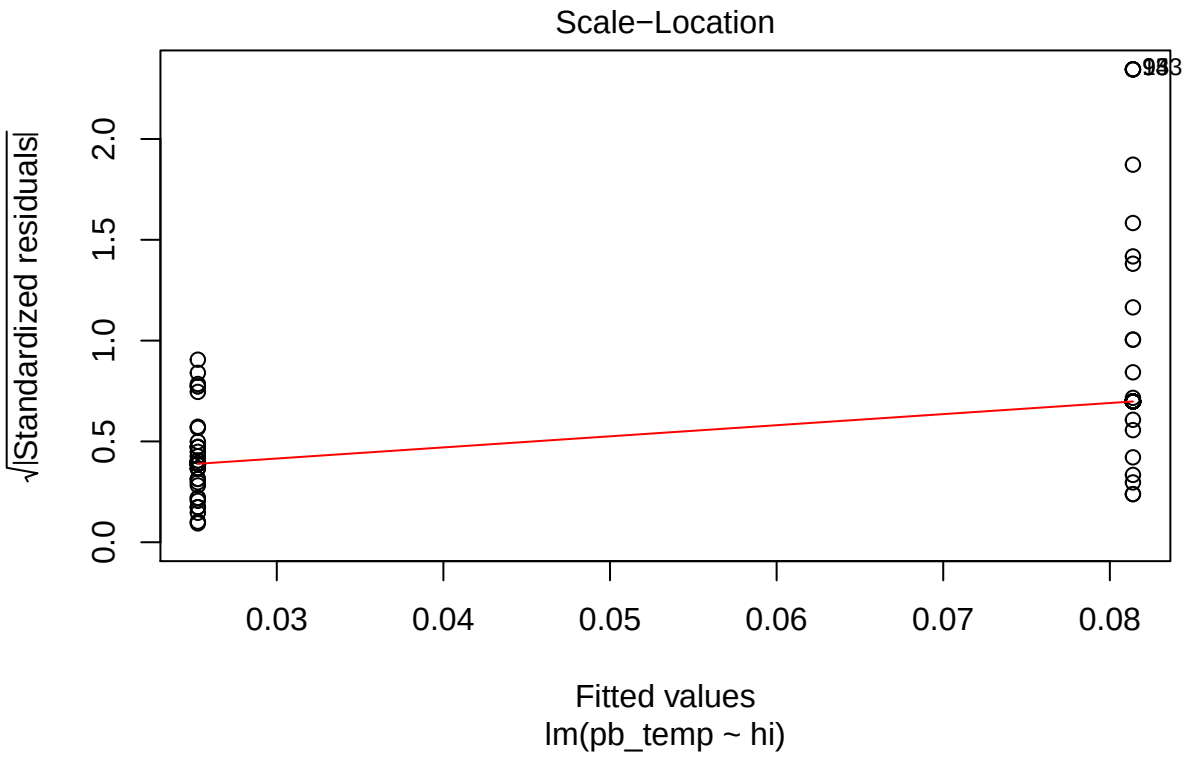

```
bptest(fit$model)
```

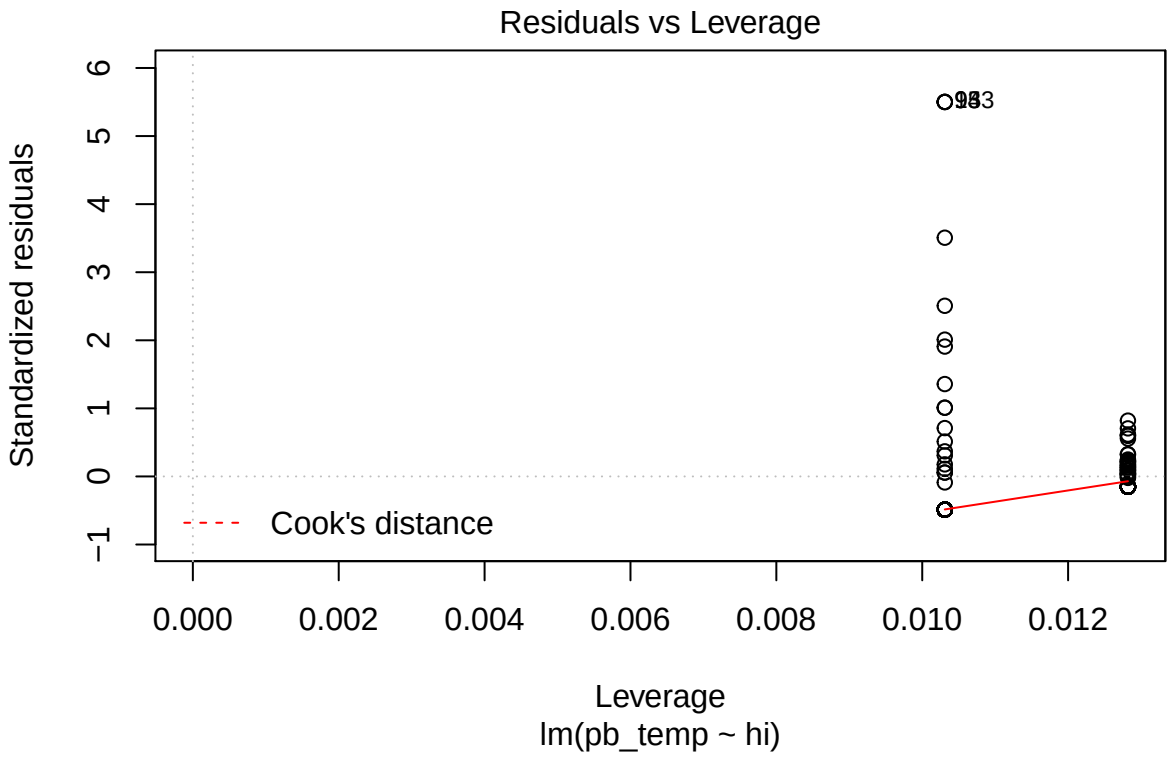
```
##
##  studentized Breusch-Pagan test
##
## data:  fit$model
## BP = 5.5242, df = 1, p-value = 0.01876
```

```
#Do the same without the outlier
dt2 <- dt[dt$black_total < 12,]
fit2 <- lm(pb_temp ~ hi, dt2)
plot(fit2)
```


Residuals vs Fitted
Residuals
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.03
0.04
0.05
0.06
0.07
0.08
Fitted values
lm(pb_temp ~ hi)


Normal Q−Q
Standardized residuals
0
1
2
3
4
5
6
−2
−1
0
1
2
Theoretical Quantiles
lm(pb_temp ~ hi)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
0.03
0.04
0.05
0.06
0.07
0.08
Fitted values
lm(pb_temp ~ hi)




```
bptest(fit2$model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  fit2$model
## BP = 5.8994, df = 1, p-value = 0.01515
```

```
fit
```

```
##
## Call:
## lm(formula = pb_temp ~ hi, data = dt)
##
## Coefficients:
## (Intercept)       hiHigh
##     0.08139     -0.05138
```

```
fit2
```

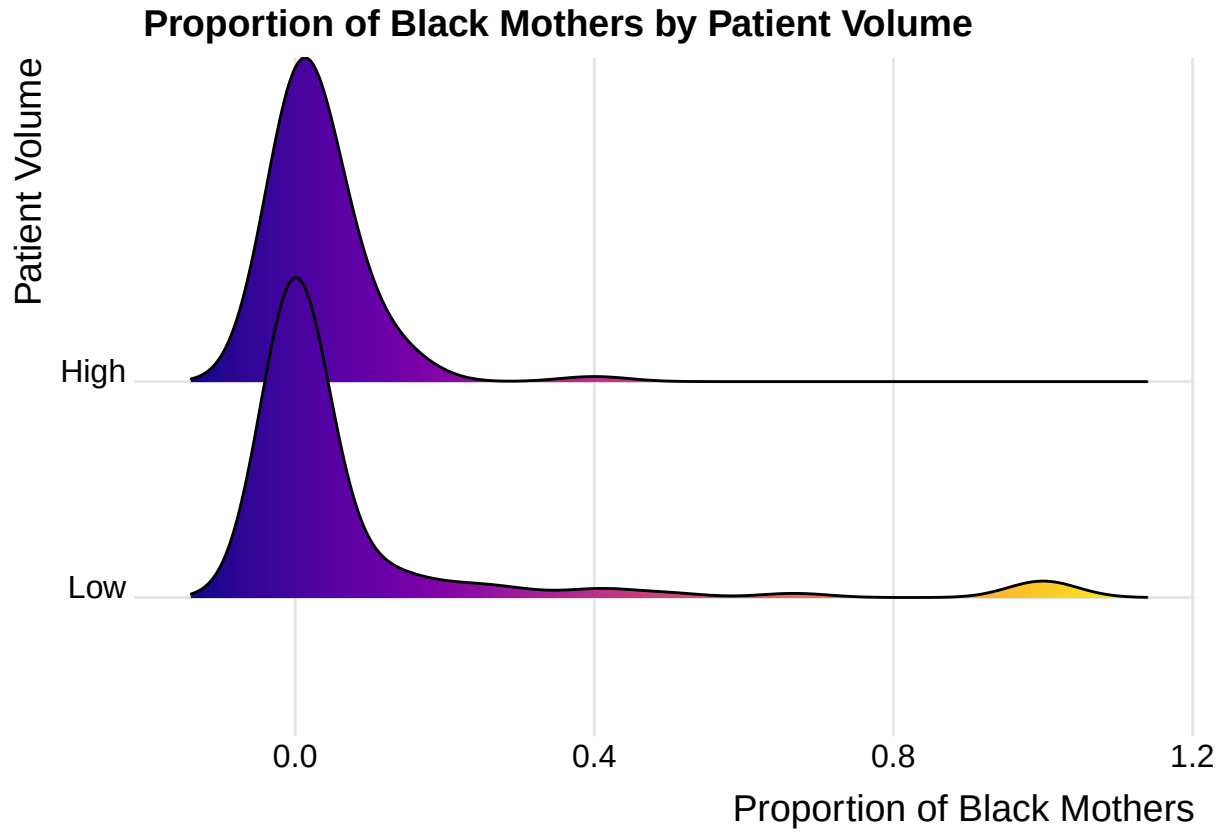
```
##
## Call:
## lm(formula = pb_temp ~ hi, data = dt2)
##
## Coefficients:
## (Intercept)       hiHigh
##     0.08139     -0.05613
```

Removing the outliers makes little difference in either the results or the plots. The scale variable, percentage

of mothers who are black, has a strange distribution of residuals. It may need transformation to give back valid results; or it may not be possible to use a linear regression on these data. That possibility is more fully evaluated below.

Since there are enough cases, we can use a ridge plot to see the distributions for each subset of data by the factor variable.

```
library(ggplot2)
library(ggridges)
library(viridis)


x = dt$pb_temp
y = dt$hi

draw_ridges <- function(dt, x, y, title) {
  ggplot(dt, aes(x, y, fill = ..x..)) +
          geom_density_ridges_gradient(scale = 1.5,
                                       show.legend = FALSE) +
          scale_fill_viridis(option = "C") +
          theme_ridges() +
          labs(title = title,
               x = "Proportion of Black Mothers",
               y = "Patient Volume")
}
draw_ridges(dt, x, y, "Proportion of Black Mothers by Patient Volume")
```

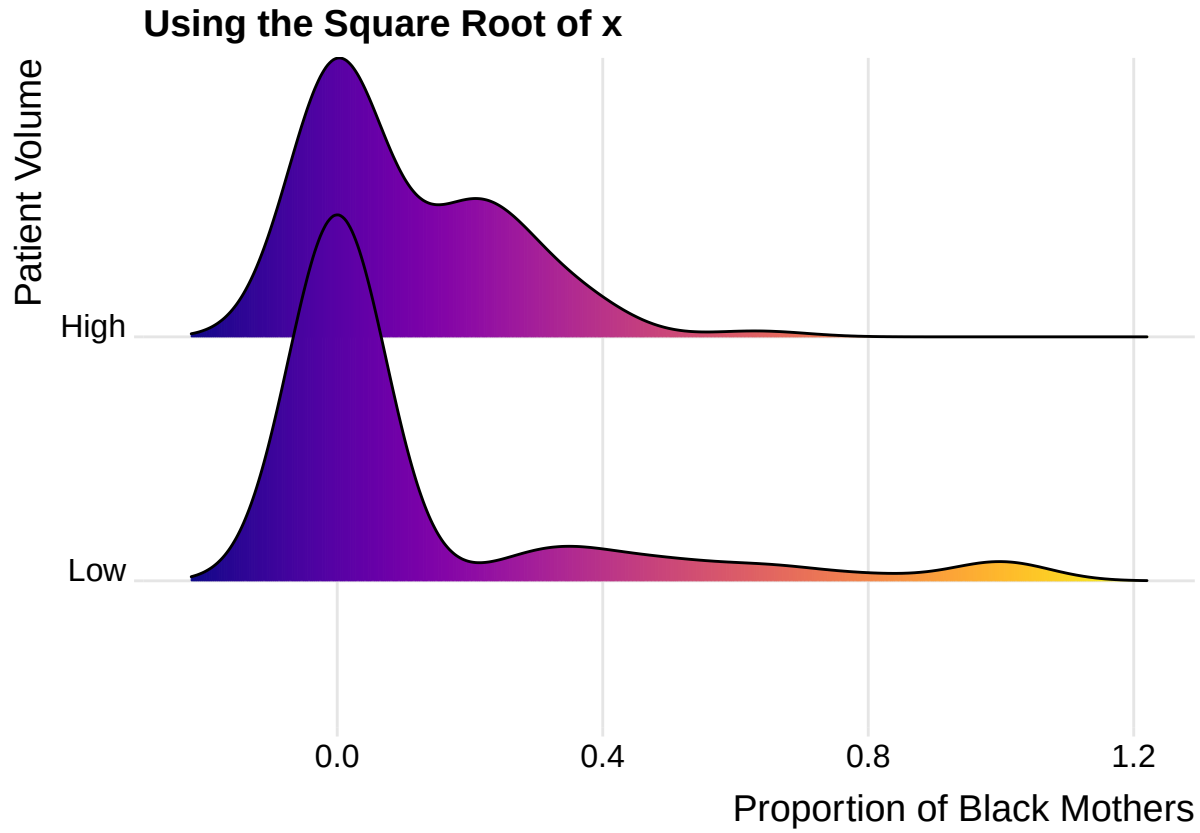


The plot above gives an excellent notion of how high-volume and low-volume doctors compare with regard to the proportion of their patients who are black mothers. In order to answer the research question, we need to either transform the data or use a non-parametric method.

Transformation of x to sqrt(x)

```
T_x <- sqrt(x)
draw_ridges(dt, T_x, y, "Using the Square Root of x")
```

```
## Picking joint bandwidth of 0.0733
```



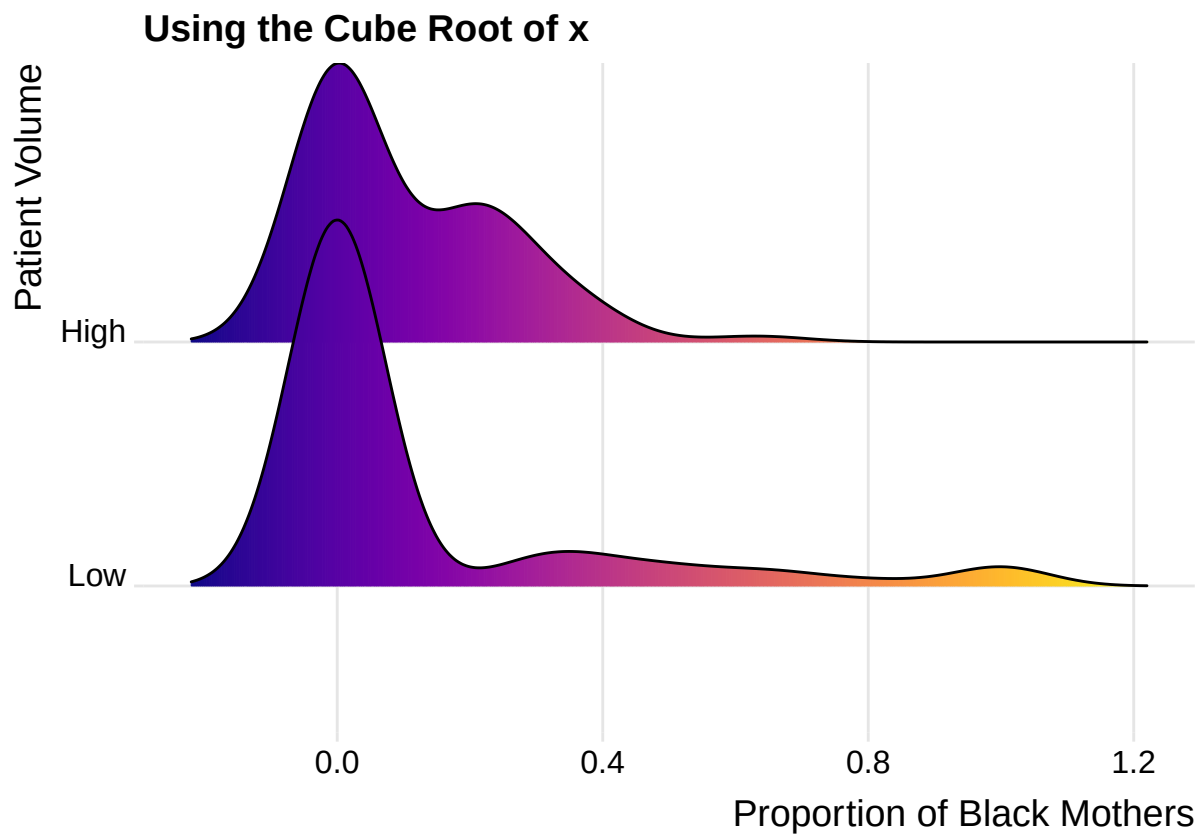
This is no improvement.

Transformation of x to its Cube Root

Trying a cube root transformation:

```
c_x <- sign(x) * abs(x) ^ 1/3 #Avoid complex numbers for some roots
draw_ridges(dt, T_x, y, "Using the Cube Root of x")
```

```
## Picking joint bandwidth of 0.0733
```



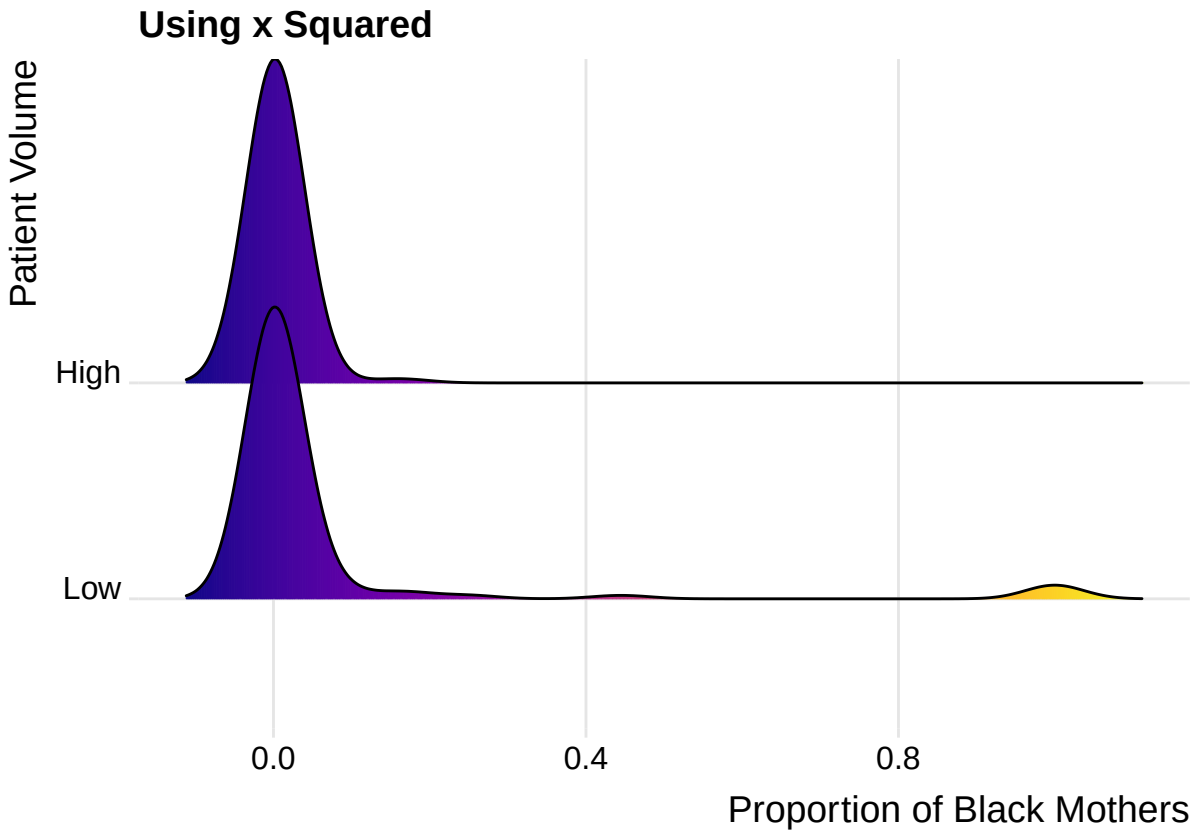
The cube root has little to recommend it over the square root.

A log transformation won't work because we'll get infinity as answers for some transformations.

Transformation of x to *x^2*

```
sq_x <- x^2
draw_ridges(dt, sq_x, y, "Using x Squared")
```

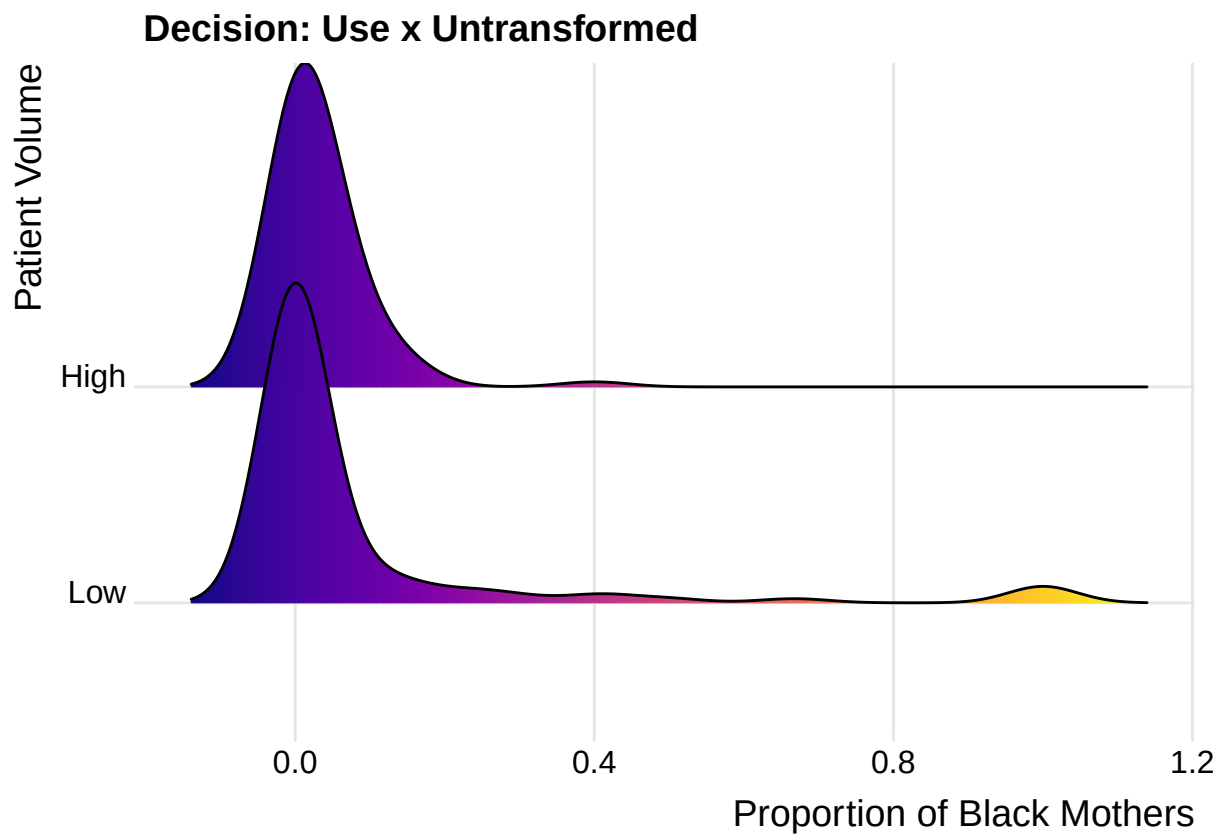
```
## Picking joint bandwidth of 0.0372
```



Going with x untransformed for now, and I will include a nonparametric test.

```
draw_ridges(dt, x, y, "Decision: Use x Untransformed")
```

```
## Picking joint bandwidth of 0.0464
```



The Model

Here again is the code for a linear model of proportion of black mothers predicted by the volume of births (high vs. low) attended by each physician.

```
fit <- lm(x ~ y, dt)
bp <- bptest(fit$model)
bp
```

```
##
##  studentized Breusch-Pagan test
##
## data:  fit$model
## BP = 5.5242, df = 1, p-value = 0.01876
```

```
fit$coef
```

```
## (Intercept)       yHigh
##  0.08138660 -0.05138153
```

```
slope <- round(fit$coef[[2]],3)
intercept <- round(fit$coef[[1]],3)
summaryfit <- summary(fit)
af <- anova(fit)
af
```

```
## Analysis of Variance Table
```

```
##
## Response: x
##            Df Sum Sq  Mean Sq F value  Pr(>F)
## y           1 0.1149 0.114948  3.9893 0.04735 *
## Residuals 174 5.0137 0.028814
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
p <- round(af[5],3)
F <- round(af[4],3)
```

This model is useless because one of its assumptions, homoscedasticity, is violated (the Breusch-Pagan test has a p of 0.0187557, so we reject the null hypothesis of homoscedasticity).

But just to go through it: Beta is -0.051, which is also the slope, and the intercept is 0.081. The slope is very slight and it is negative. According to the ANOVA, it is signficant ($F = 3.989$, p = 0.047), but only at alpha $< .05$. Interpretation: Low-volume doctors attend the births of a greater proportion of black mothers than their high-volume counterparts do. Besides the lack of homoscedasticity, what I distrust about using this model to draw major conclusions is the low amount of variance it accounts for: only 2.24%.

T tests

I also looked at two t tests–one for the difference between means and the other for the difference between medians–because these analyses are robust with non-normality and the linear regression model fails to account for a sufficient amount of the variance to be reliable.

```
t_tests <- function(x, y) {
  t <- t.test(x ~ y)
  t
  tp <- round(t$p.value, 3)
  # Best test:

  w <- wilcox.test(x ~ y)
  w
  wp <- round(w$p.value, 3)
  return(c(c(t, tp), c(w, wp)))
}
mytlist <- t_tests(x, y)
tp <- round(mytlist[[3]], 7)
wp <- round(mytlist[[13]], 7)
tmethod <- mytlist[[8]]
wmethod <- mytlist[[16]]
```

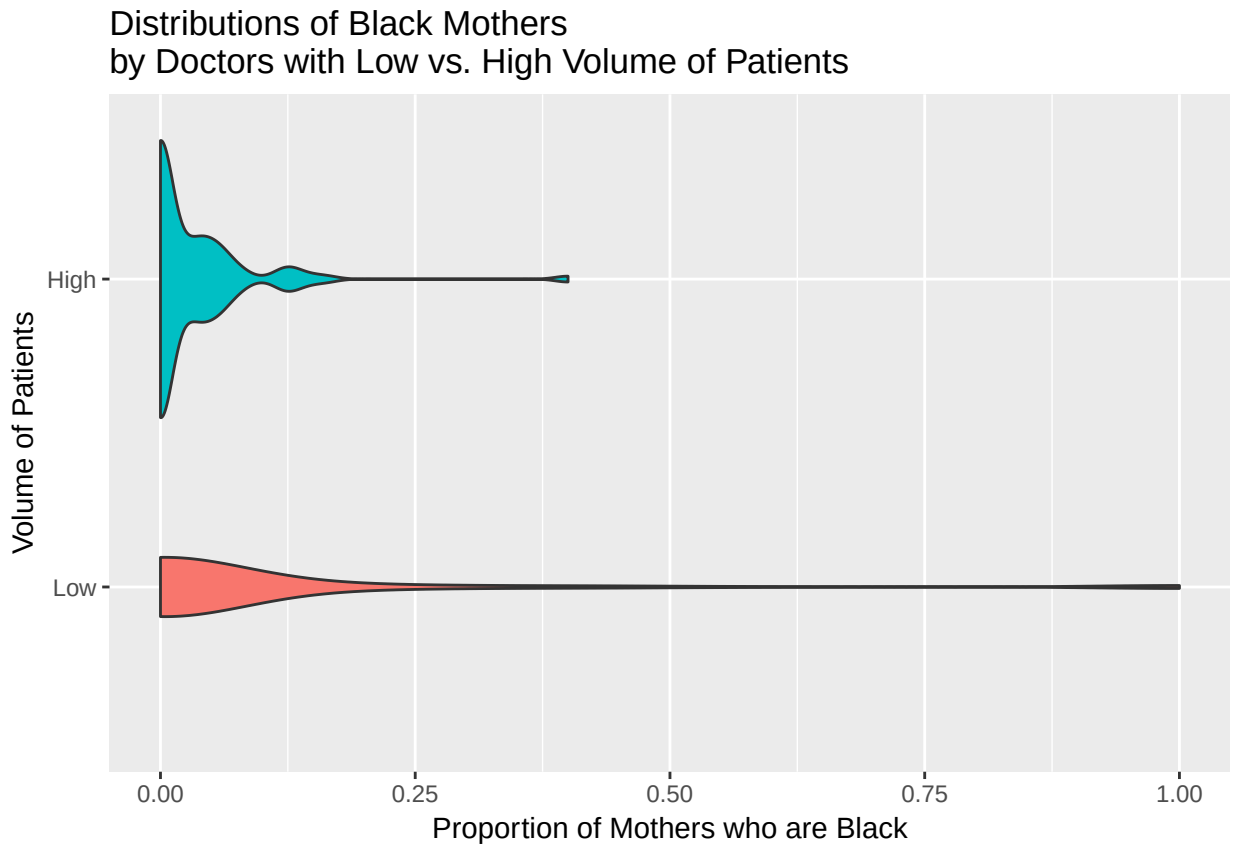
The analysis is not clear because the two t tests disagree at alpha = .05. The Welch Two Sample t-test has p = 0.0309229 and the Wilcoxon rank sum test with continuity correction has p = 0.0706971.

Conclusion

Being a low-volume doctor predicts a greater proportion of black mothers. I'm not convinced that looking at this one predictor is a good idea, at least if we want to understand how, from a phyisician's point of view, the volume of patients and the proportion of black patients relate to each other. There is too much unexplained variance to be happy with the model, so I also include some t tests.

The best image for understanding the data is this one:

```
myViolin <- function(dt, title) {
  ggplot(dt,
         aes(x = hi, y = pb_temp, fill = hi)) +
          geom_violin(show.legend = FALSE) +
          labs(title = title,
               x = "Volume of Patients",
               y = "Proportion of Mothers who are Black") +
          coord_flip()
}
title <- "Distributions of Black Mothers\nby Doctors with Low vs. High Volume of Patients"
myViolin(dt, title)
```



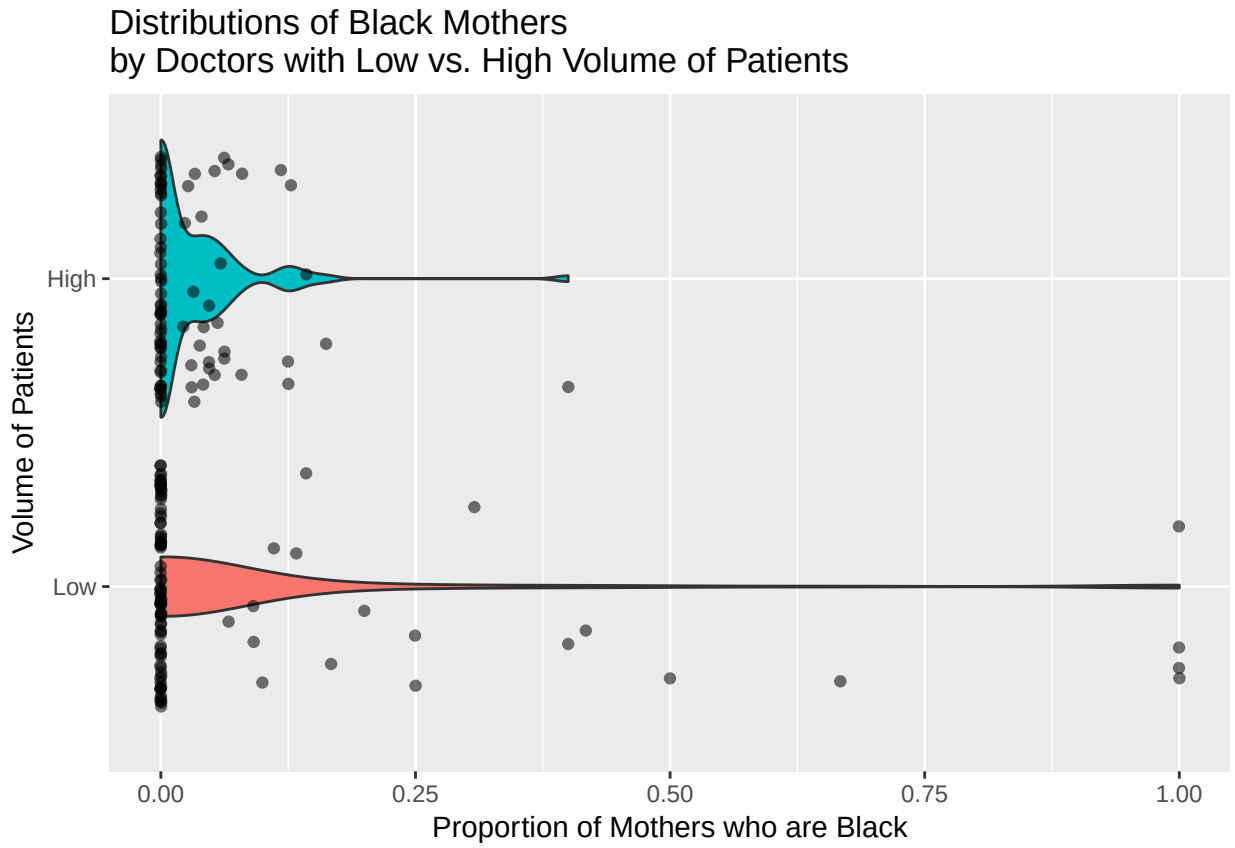

```
tbl <- table(dt$hi)
low_volume <- tbl[1]
high_volume <- tbl[2]
```

The number of low-volume doctors is 97 and the number of high-volume doctors is 79.

This next plot is not as neat, but it gives a sense for the number of doctors represented at each point along the violin plots. Each dot represents a doctor. There are 97 (low) + 79 (high) = 176 doctors.

```
myJitteredViolin <- function(dt, title) {
  ggplot(dt,
         aes(x = hi, y = pb_temp, fill = hi), show.legend = FALSE) +
          geom_violin(show.legend = FALSE) +
          labs(title = title,
               x = "Volume of Patients",
```

```
              y = "Proportion of Mothers who are Black") +
          coord_flip() +
          geom_jitter(aes(alpha = .15), show.legend = FALSE)
}
title <- "Distributions of Black Mothers\nby Doctors with Low vs. High Volume of Patients"
myJitteredViolin(dt, title)
```

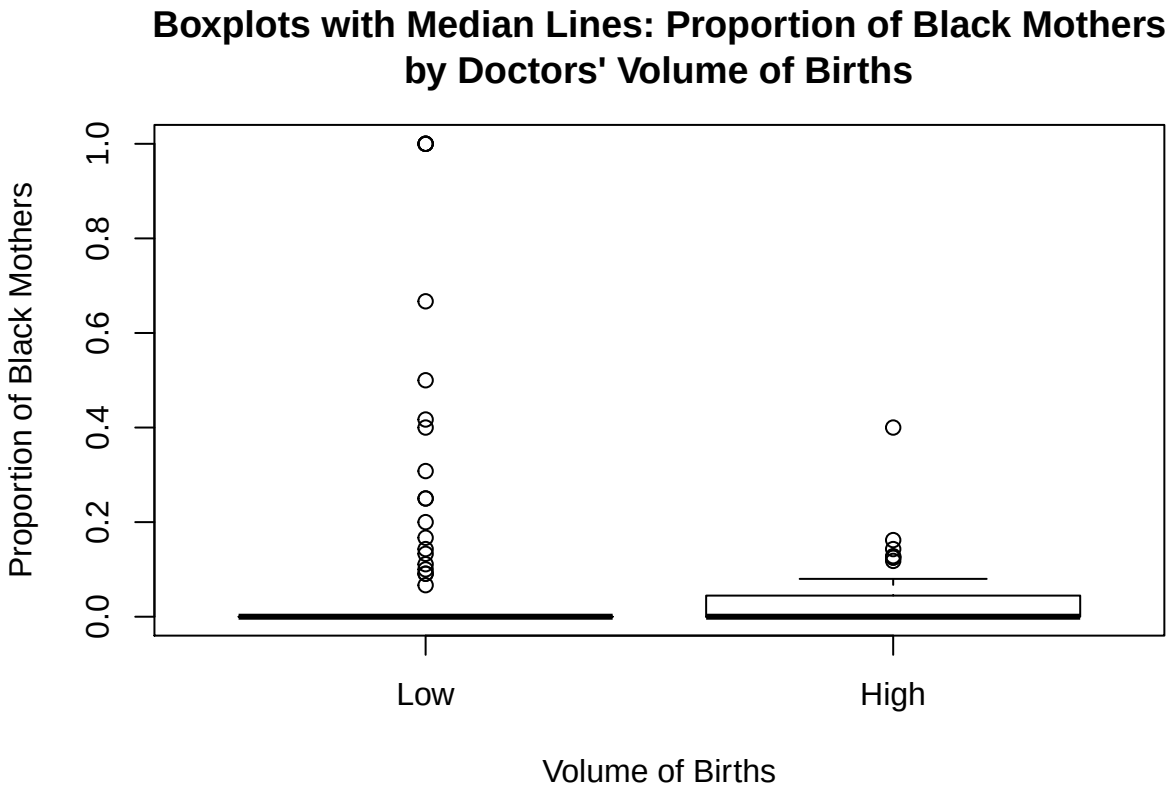



```
#A little extra helpful information to consider with the plots.
extract_meanmeds <- function(dt) {
  lows <- dt[dt$hi %in% "Low",]
  highs <- dt[dt$hi %in% "High",]
  lmean <<- mean(lows$pb_temp)
  hmean <<- mean(highs$pb_temp)
  lmed <<- median(lows$pb_temp)
  hmed <<- median(highs$pb_temp)
  lvar <- sd(lows$pb_temp)
  hvar <- sd(highs$pb_temp)
  return(c(lmean, hmean, lmed, hmed, lvar, hvar))
}
extract_meanmeds(dt)
```

```
## [1] 0.08138660 0.03000506 0.00000000 0.00000000 0.22256689 0.05753740
```

The t tests show that the means and medians are different, but they do not show which is higher. I added a little extra code above to get those (non-transformed) numbers. The mean proportion of black mothers for low-volume doctors is 0.0813866 and the median is almost undetectably above 0. For high-volume doctors, the mean is 0.0300051 and again, the median is almost undetectably above 0.

```
boxplot(x ~ y,
        main = "Boxplots with Median Lines: Proportion of Black Mothers\nby Doctors' Volume of Births",
        ylab = "Proportion of Black Mothers",
        xlab = "Volume of Births")
```



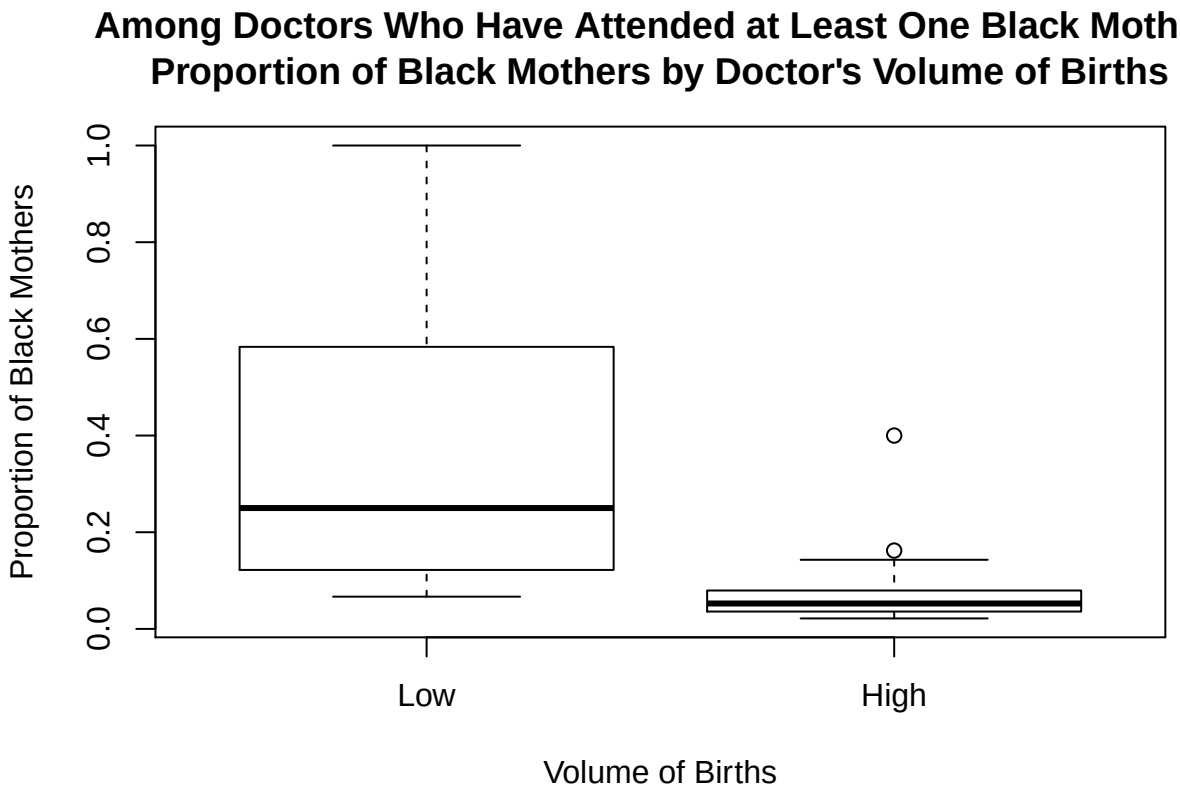
The non-paramaetric t test–the Wilcoxon Rank Sum test–has a lot going for it: it is oblivious to the power of outliers because it uses the "rank" of each data point rather than its raw value. Rank cannot be influenced by the distance of any points from each other; all that matters is that one is greater or less than the other. Despite the fact that the Wilcoxon Rank Sum test does not "predict" the proportion of black patients, then, it is the analysis of choice for this data set with its extreme values and its somewhat multi-modal distribution.

A Further Examination

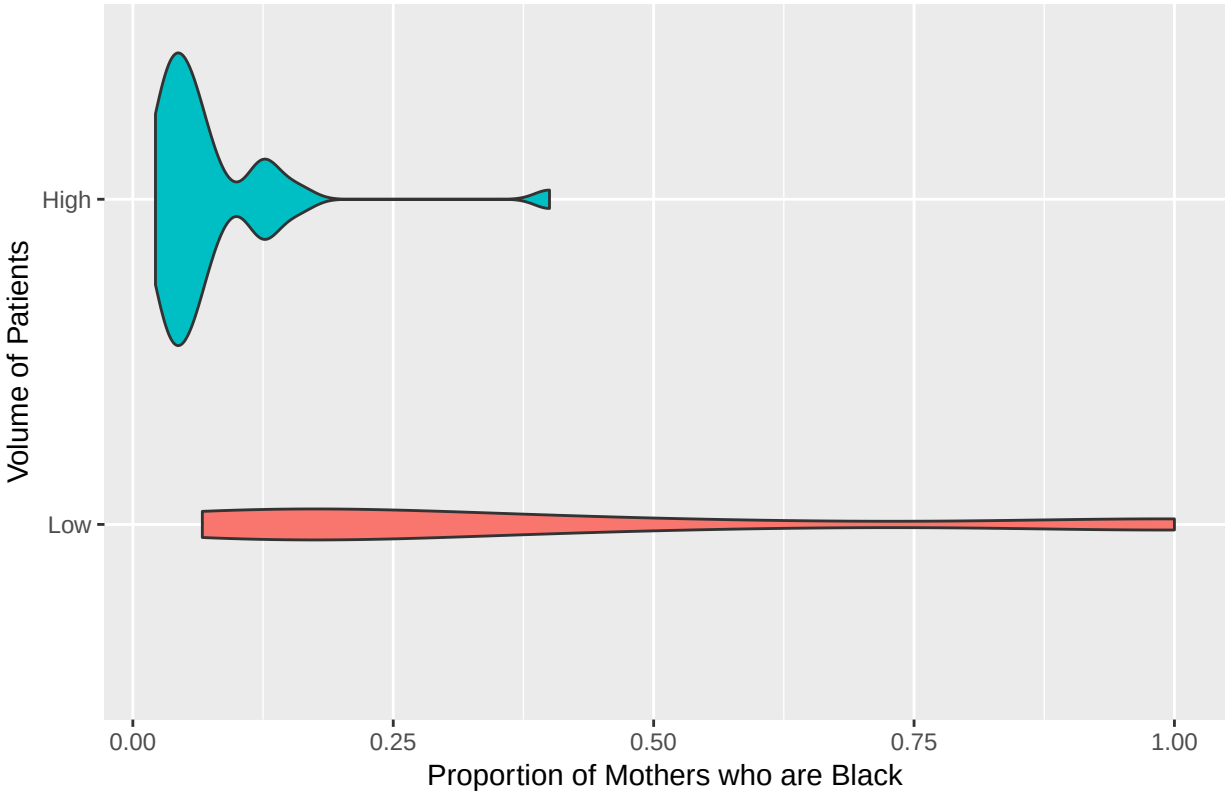
What if the Question 5 were slightly different: Are high volume doctors who have any black mothers as patients more likely to have a higher proportion of black patients than low volume doctors who have any black mothers as patients?

```
dt3 <- dt %>%
        dplyr::select(att_id, hi, pb_temp) %>% #select the vars we need
        filter(pb_temp > 0) #select the cases we want

x <- dt3$pb_temp #store in x & y for ease of coding
y <- dt3$hi
boxplot(x ~ y, #take a quick look at the medians
        main = "Among Doctors Who Have Attended at Least One Black Mother:\nProportion of Black Mothers
```

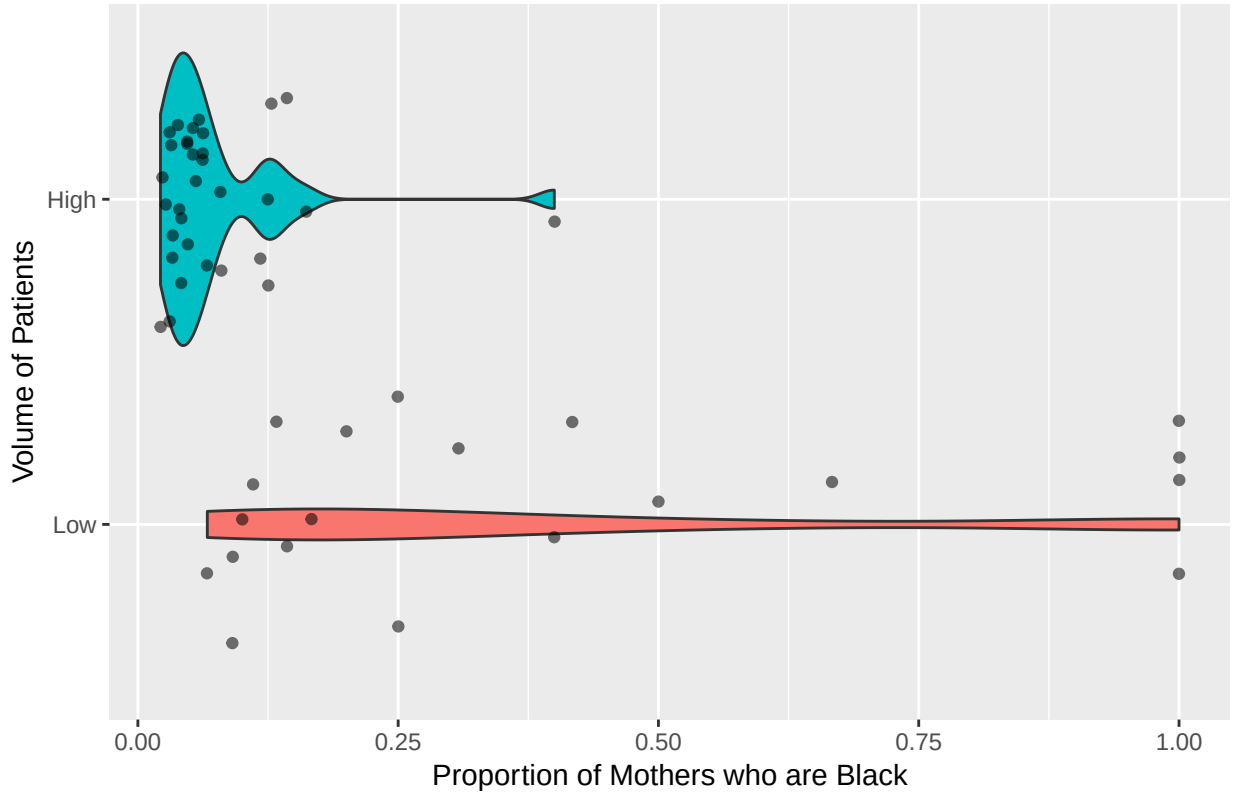
```
        ylab = "Proportion of Black Mothers",
        xlab = "Volume of Births")
```




```
myViolin(dt3, "") #take a better look
```




```
myJitteredViolin(dt3, "") #get more detail
```




```
v <- extract_meanmeds(dt3) #get summary values
lmean <- v[1]
lmed <- v[2]
hmean <- v[3]
hmed <- v[4]
lsd <- v[5]
hsd <- v[6]
```

Removing doctors who have never attended a black woman giving birth leaves 52 doctors. This revised dataset gives less information about the distribution of black women among doctors who deliver babies in a state in the US. However, it also takes away the dramatic influence in means and medians of the fact that the majority of doctors in the original dataset have never attended a black woman's lying-in.

What we can see clearly from this revised data set is that high-volume doctors attend few black mothers, wherease low-volume doctors are spread from 2.17% of their patients being black to 100%. The standard deviation for high-volume doctors is 0.0704272 and for low-volume doctors it is 0.3465475. Hence doing any kind of analysis that relies heavily on variance comparison would be unwise. One of the assumptions in OLS regression is that the residuals have the same variance. Here they do not seem to. The OLS regression model below is followed by a test for heteroscedasticity.

Comparing the two types of t test (means and medians comparisons) is informative.

```
library(lmtest)

mytlist <- t_tests(x, y)
```

```
## Warning in wilcox.test.default(x = c(0.111, 0.667, 0.133, 0.1, 0.5,
## 0.143, : cannot compute exact p-value with ties
```

```
tp <- round(mytlist[[3]], 7)
wp <- round(mytlist[[13]], 7)
tmethod <- mytlist[[8]]
wmethod <- mytlist[[16]]
fit <- lm(x ~ y, dt3)
bp <- bptest(fit$model)
bp
```

```
##
##  studentized Breusch-Pagan test
##
## data:  fit$model
## BP = 15.403, df = 1, p-value = 0.00008687
```

```
bp_p <- bp$p.value
sfit <- summary(fit)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: x
##           Df Sum Sq Mean Sq F value      Pr(>F)
## y          1 1.2654 1.26543  25.978 0.000005315 ***
## Residuals 50 2.4356 0.04871
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Both t tests have $p < .001$: 0.0005767 for the Welch Two Sample t-test and 0.0000003 for the Wilcoxon rank sum test with continuity correction.

Means are 0.394725 for low-volume doctors and 0.25 for high-volume doctors.

Medians are 0.074075 for low-volume doctors and 0.0526 for high-volume doctors.

The linear model is useless because the null hypothesis of studentized Breusch-Pagan test is that the residuals have constant variance. We reject the null based on the p of 0.0000869. It's too bad, because if we could trust it, it would explain 32.9% of the variance.

Again, the test of choice for these data, even reduced, is the Wilcoxon rank sum test with continuity correction, which confirms that among doctors in a state in the US who have ever attended a black woman giving birth, low-volume doctors attend a greater median percentage (7.41% compared to 5.26% among high-volume doctors).

Deliverable Data Set

The researcher wants the data set of all doctors delivering babies in a state in the US with a few specific variables. The data are extracted and the file saved in the next code chunk.

```
dt <- dt %>% dplyr::select(att_id, num_births, cx_total, perc_black) %>%
        rename(num_cx = cx_total) %>%
        ungroup()
#put the var names back to uppercase
names(dt) <- toupper(names(dt))

#save the file
write.csv(dt, "Physicians Birth Race Data.csv")
```