Project 2: Group 1 - Boston Housing Racial Bias

Will Johnson

12/3/2021

Exploring our Dataset

```r
# Download our dataset
library(mlbench)
data(BostonHousing2)
data <- BostonHousing2
head(data)
```

```
##         town tract      lon     lat medv cmedv    crim zn indus chas   nox
## 1     Nahant  2011 -70.9550 42.2550 24.0  24.0 0.00632 18  2.31    0 0.538
## 2 Swampscott  2021 -70.9500 42.2875 21.6  21.6 0.02731  0  7.07    0 0.469
## 3 Swampscott  2022 -70.9360 42.2830 34.7  34.7 0.02729  0  7.07    0 0.469
## 4 Marblehead  2031 -70.9280 42.2930 33.4  33.4 0.03237  0  2.18    0 0.458
## 5 Marblehead  2032 -70.9220 42.2980 36.2  36.2 0.06905  0  2.18    0 0.458
## 6 Marblehead  2033 -70.9165 42.3040 28.7  28.7 0.02985  0  2.18    0 0.458
##      rm  age    dis rad tax ptratio      b lstat
## 1 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
## 6 6.430 58.7 6.0622   3 222    18.7 394.12  5.21
```

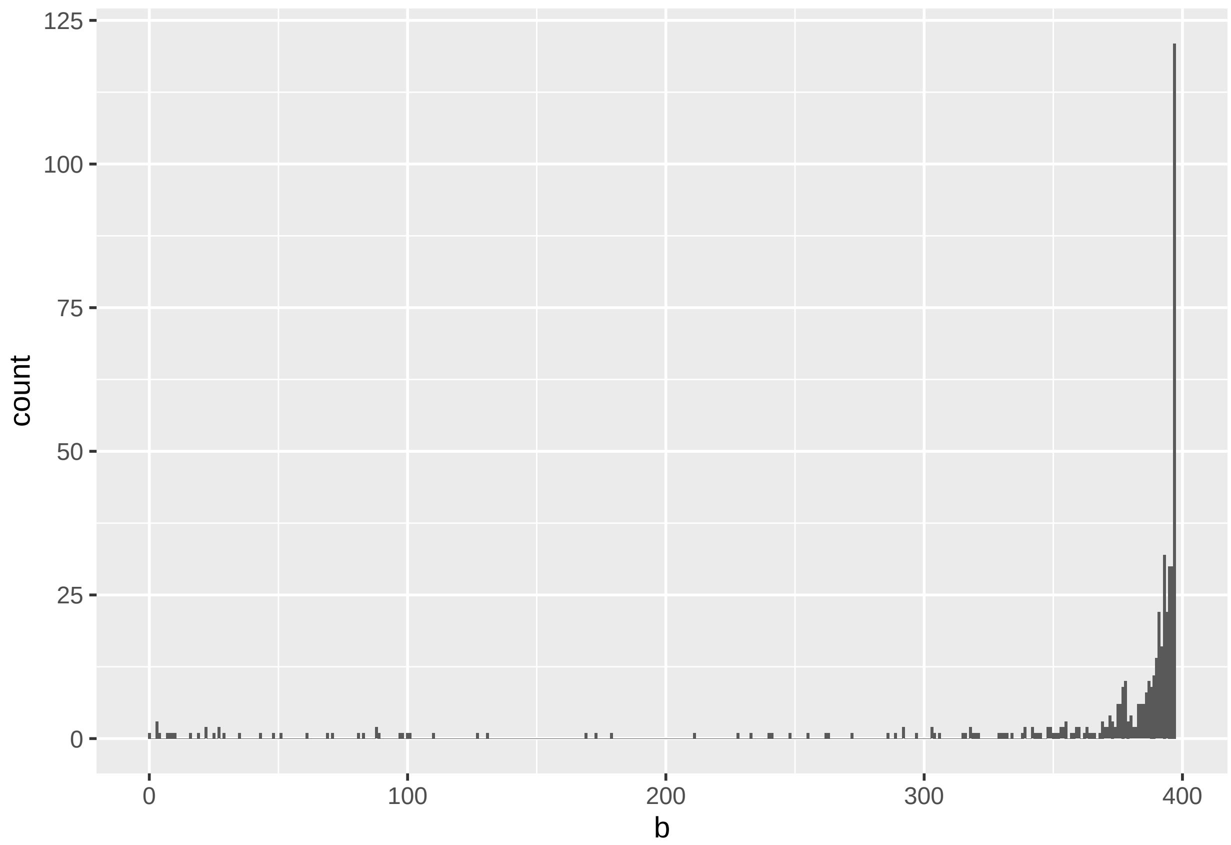
"Majority Black" response variable

The variable "b" is described in the census documents as…

> 1000(B - 0.63)^2 where B is the proportion of blacks by town

Originally, we had planned to set this as a "majority black" binary variable. Based on the description, we thought the values would be between 0 and 1, or possibly 0 and 100. But the variable itself was a strange one instead. Take a look at the distribution of this variable `b`.

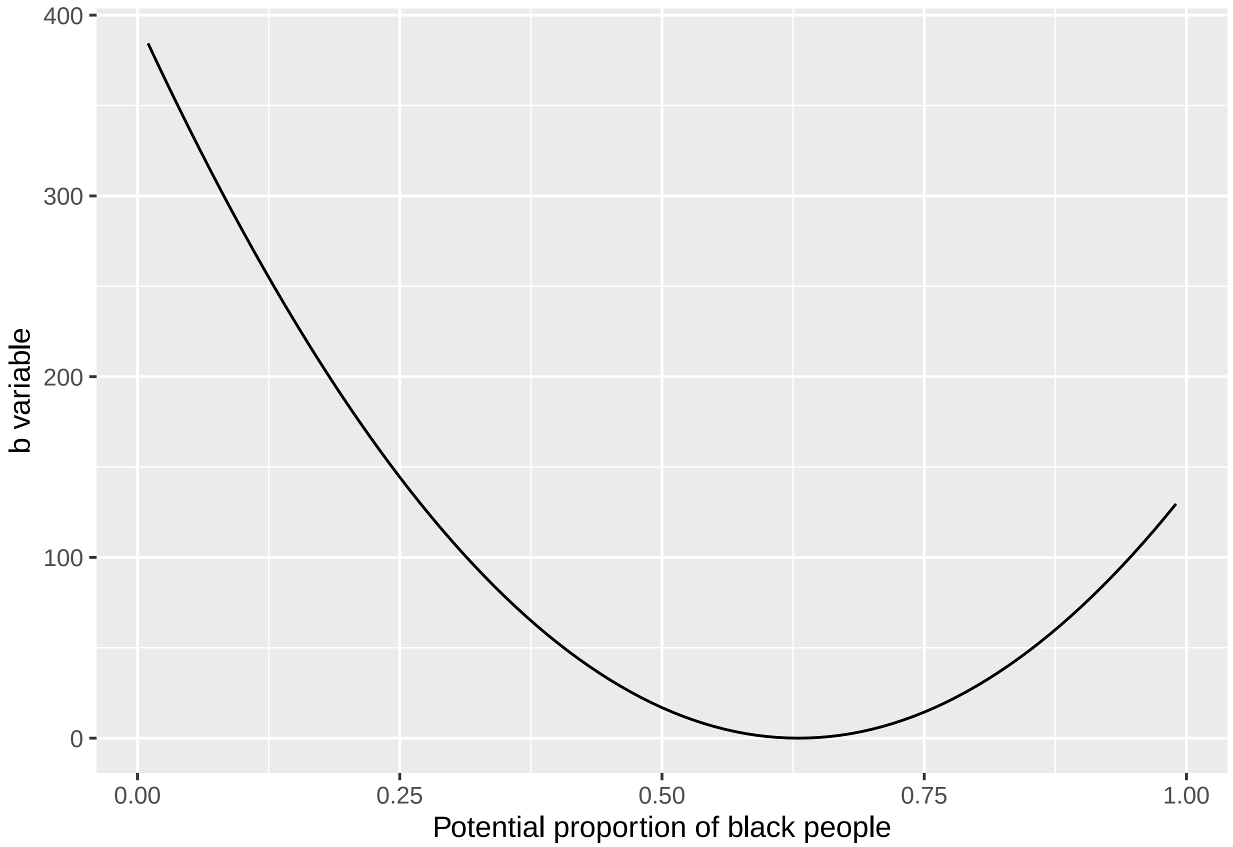
```r
# plot the distribution on the "b" variable.
ggplot(data, aes(x = b))+
  geom_histogram(binwidth = 1)
```



Not only is the range very odd (0 to 400), the majority of these towns are in the high-end, suggesting that the data has been transformed so that the higher the value is, the whiter the neighborhood.

The most interesting part about our findings is that if we try to reverse engineer the original B value (based on their seemingly arbitrary formula b=1000(B-.63)2), it's not possible to get the answer. The result of the previously-mentioned formula is a quadratic curve, plotted below:

```r
# Using the dataset's documentation, plot potential values for proportions of black people
# against each value of `data$b`.
props <- c()
i <- .01
for(x in 1:99){
    props <- c(props,1000*(i-.63)**2 )
    i = i + .01
}
parab <- data.frame(props,seq(.01, .99, by=.01))
colnames(parab) = c('x','y')
ggplot(data = parab, aes(x=y, y=x ))+
  geom_line()+
  labs(x = "Potential proportion of black people", y='b variable')
```



Looking at this, it appears that the dataset is impossible to perfectly break back down into its original B variable because for some values, there are potentially 2 values of B for one value of the transformed b variable. Specifically, anything more than 26% black in a neighborhood becomes vague.

Some brief external research suggests that this transformation on the variable was set to create a pseudo-parabolic relationship to account for an initial drop in value when the proportion was too mixed-race. After 75% black, the effect was expected to rise again because of preference for a neighborhood of one's own race or "self-segregation" [Harrison,Rubinfeld, 96].

To quote an assistant professor of Mathematics at CUNY who researched this:

> Harrison and Rubinfeld appear to have decided on a threshold of 63% at which to switch the regime of price decline to price increase (i.e. a so-called "ghetto threshold") [Carlisle,1]

Given that this was their decision at the time, we'll follow the precedent they set in the 1970's. We'll assess if there is a relationship between where a home falls on that threshold to see its effect on the price of a home, as well as try to predict where on that threshold a house will fall for our logistic model.

Plugging our 26% threshold into that formula, we get:

$1000 * (0.26 - 0.63)2 = 136.9$

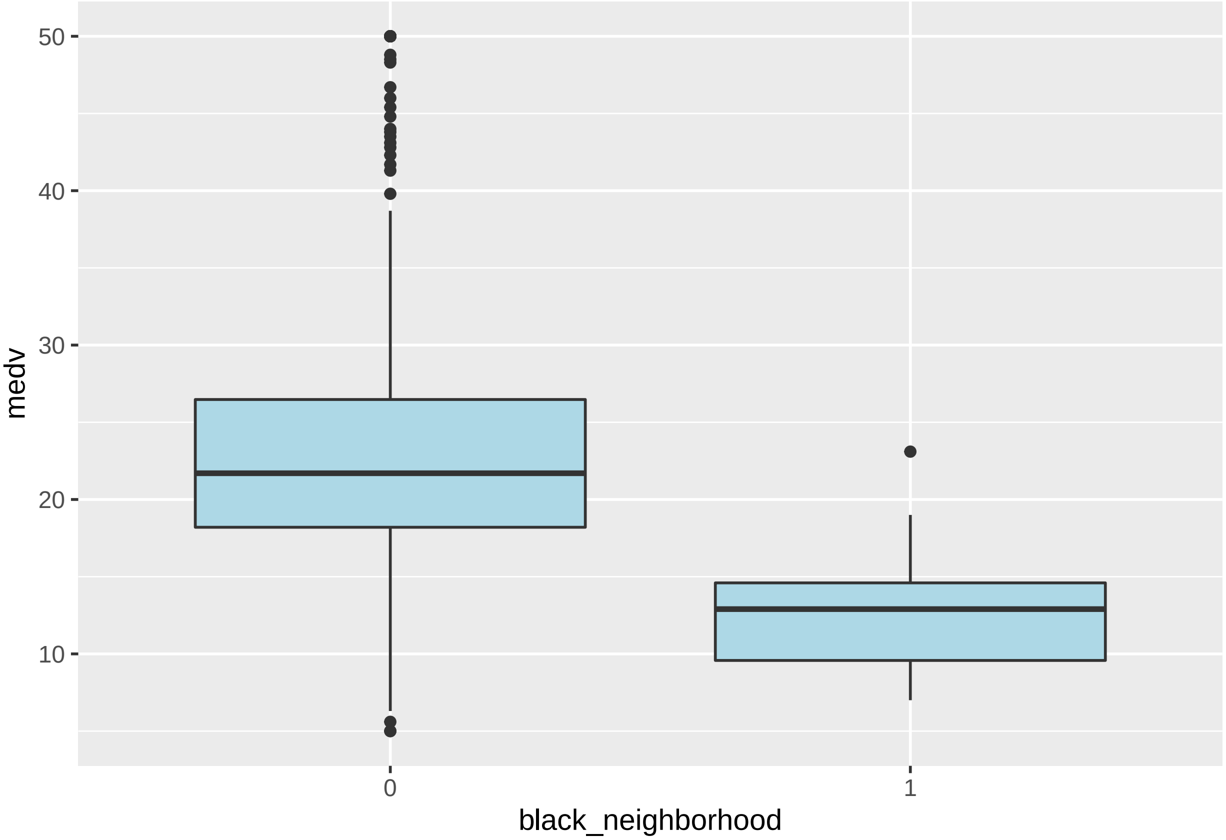
so black_neighborhood = 1 if b <= 136.9

```
# code to create binary black_neighborhood variable
data <- data %>%
           mutate(black_neighborhood = (b <= 136.9) * 1)
# what % of our neighborhoods are black neighborhoods by this definition?
sum(data$black_neighborhood)/nrow(data)
```

```
## [1] 0.07114625
```

Finally, let's chart this variable to see how some of the other's fall across it. Namely our price variable

```
data$black_neighborhood <- as.factor(data$black_neighborhood)
# plot our new variable and the price distributions across it.
ggplot(data, aes(x= black_neighborhood, y=medv))+
  geom_boxplot(fill="light blue")
```



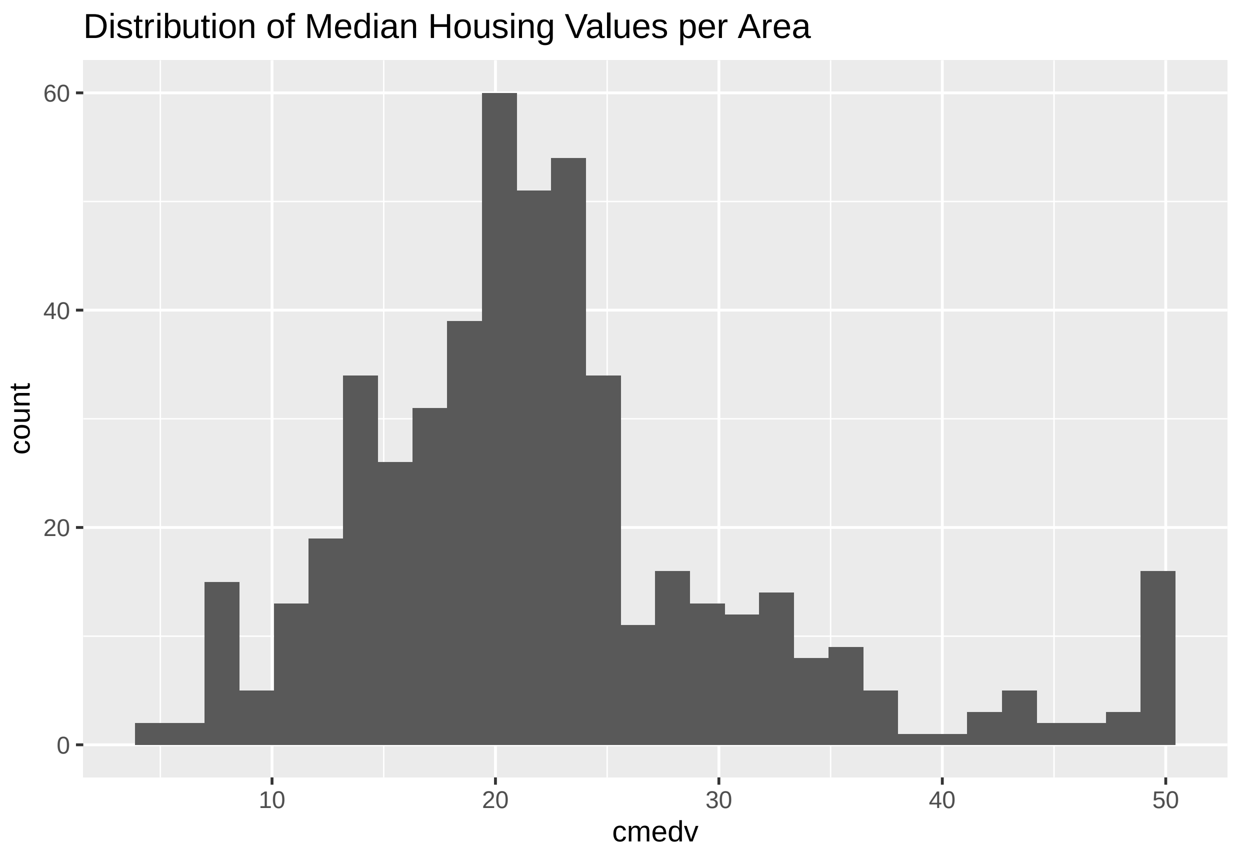
Definitely seems like there's a relationship between these two and price. Looking at it, we should assume there should be predictive value including this variable in predicting price.

Part 1 - EDA : How much does the relative blackness of a neighborhood affect price?

First we'll start by looking at a handful of other visuals to see which variables seem to affect price beyond our "black_neighborhood" variable. We'll start with looking out our prices overall.

```
ggplot(data, aes(x=cmedv))+
  geom_histogram()+
  labs(title = "Distribution of Median Housing Values per Area")
```

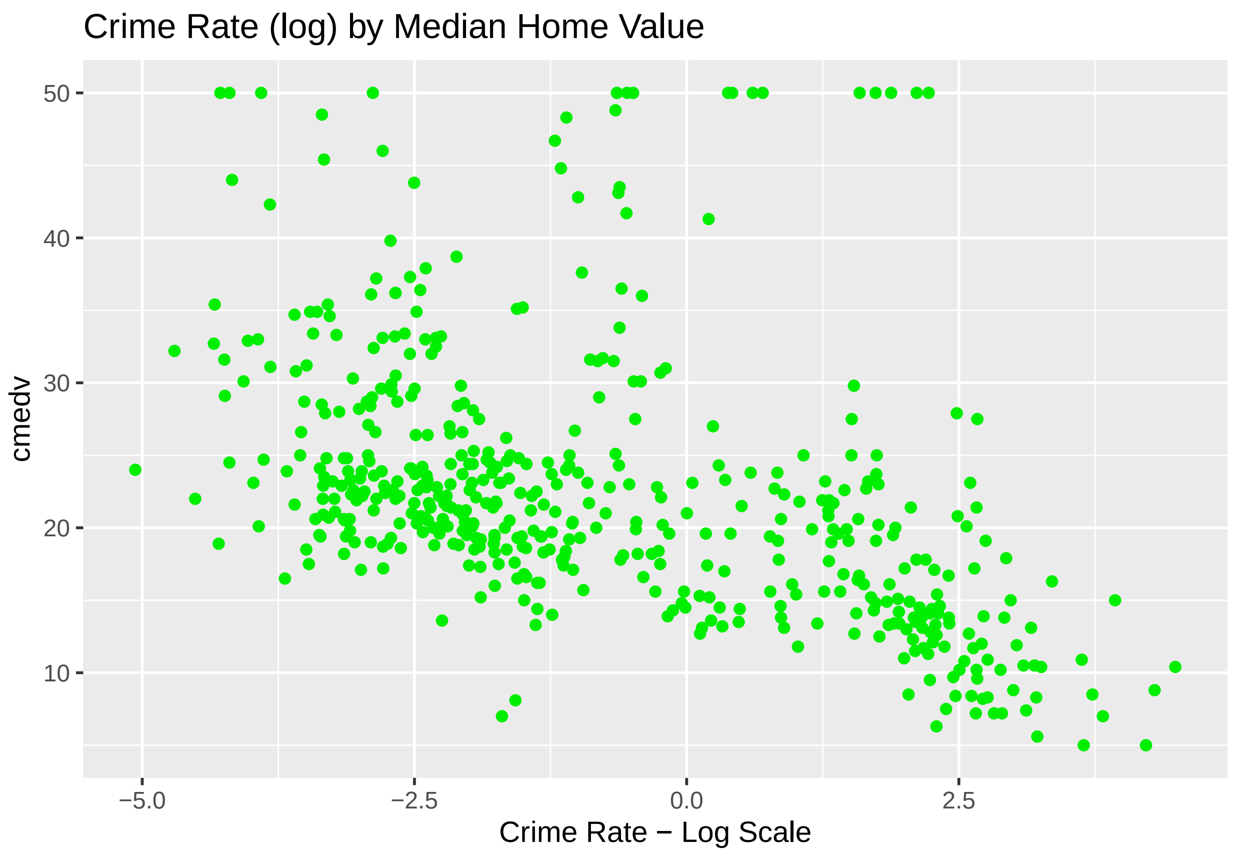
```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



Looks mostly normally distributed with a handful of towns with a particularly high median value. We should look out for those to be potential outliers later.

Next, some scatterplots between quantitative variables.

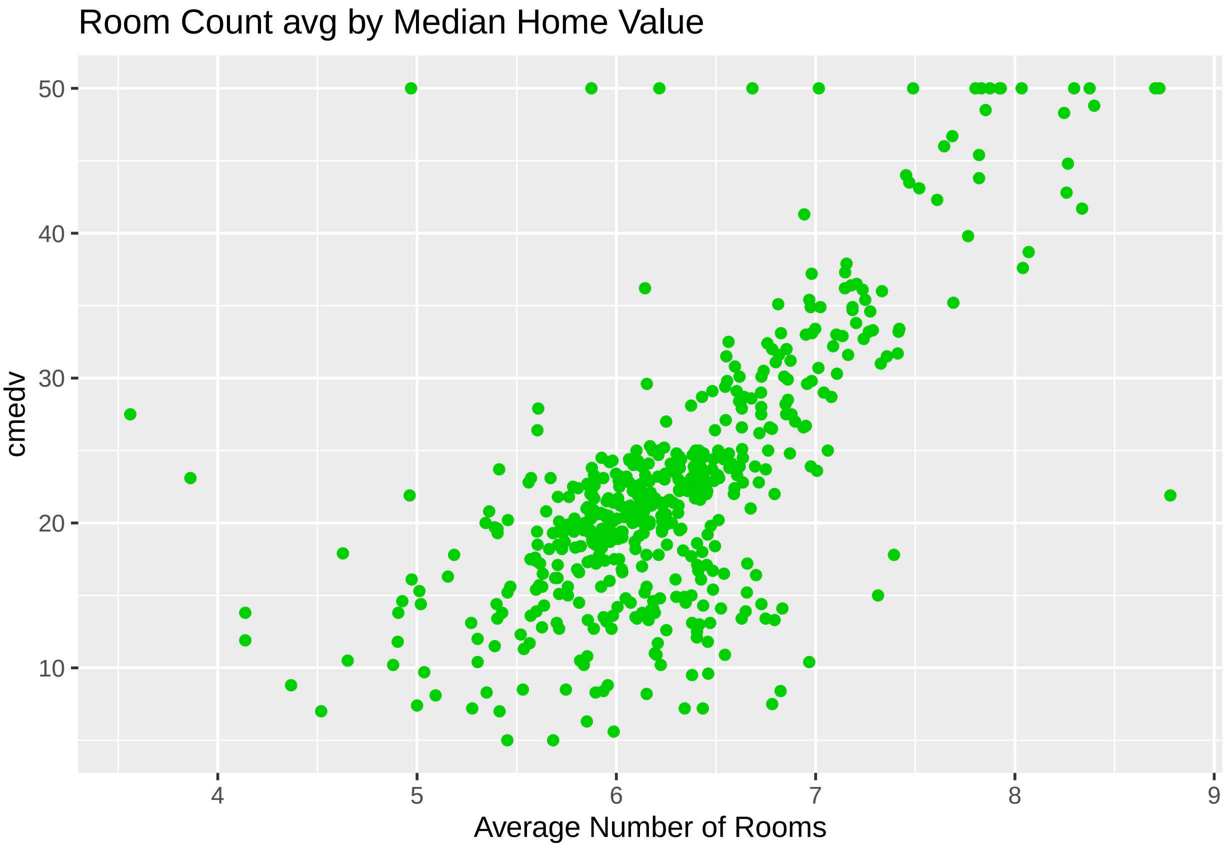
```
ggplot(data, aes(x = log(crim), y = cmedv))+
  geom_point(color = 'green2')+
  labs(title="Crime Rate (log) by Median Home Value",
       x = "Crime Rate - Log Scale")
```



Vaguely, it looks as though the more the crime rate increases, the values of houses start to decrease. We used a log scale for crime here because otherwise the majority of towns had too low a crime rate to see this trend. Oddly, there are some expensive towns that also have relatively higher crime rates per capita as well.

Next, let's see if the size of the houses make a difference based on the average number of rooms per house.

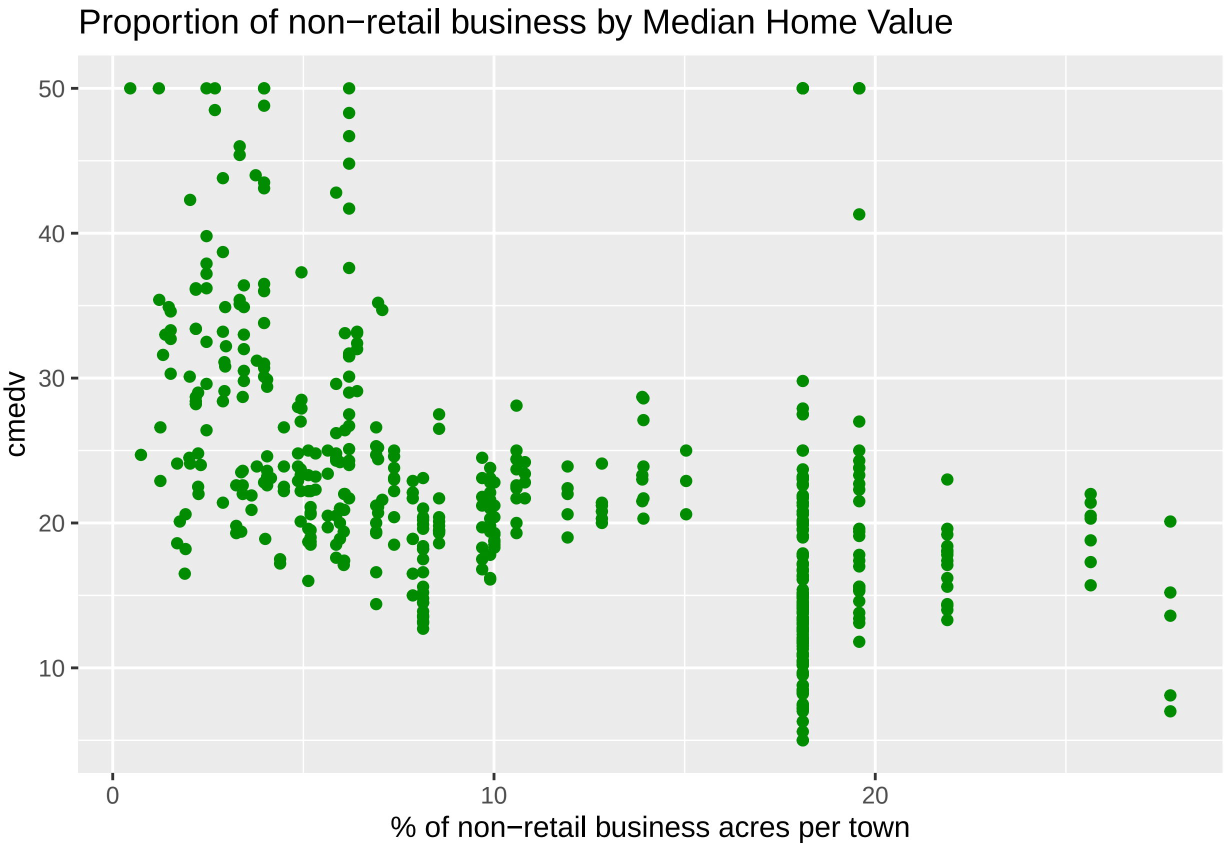
```
ggplot(data, aes(x = rm, y = cmedv))+
  geom_point(color = 'green3')+
  labs(title="Room Count avg by Median Home Value",
       x = "Average Number of Rooms")
```



Pretty clear relationship here, though again we have some odd high-leverage points here (by observation) where there are some towns where the median value is around 50,000 but the average number of rooms is far lower. Let's see if there's another variable that can account for those handful of towns that don't fit our patterns.

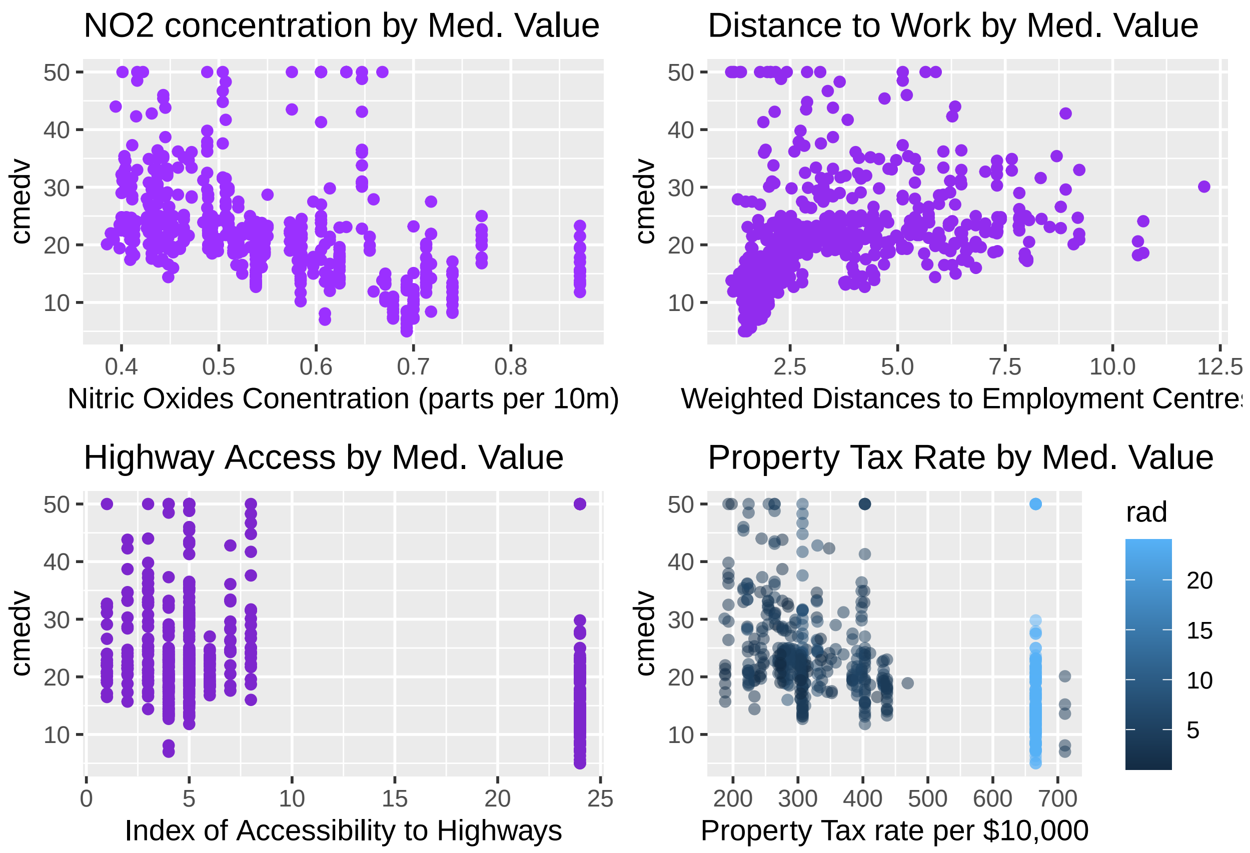
Let's try the industry proportion. Maybe things are expensive in a business neighborhood where high expenses and crime rates can happen simultaneously.

```
ggplot(data, aes(x = indus, y = cmedv))+
  geom_point(color = 'green4')+
  labs(title="Proportion of non-retail business by Median Home Value",
       x = "% of non-retail business acres per town")
```

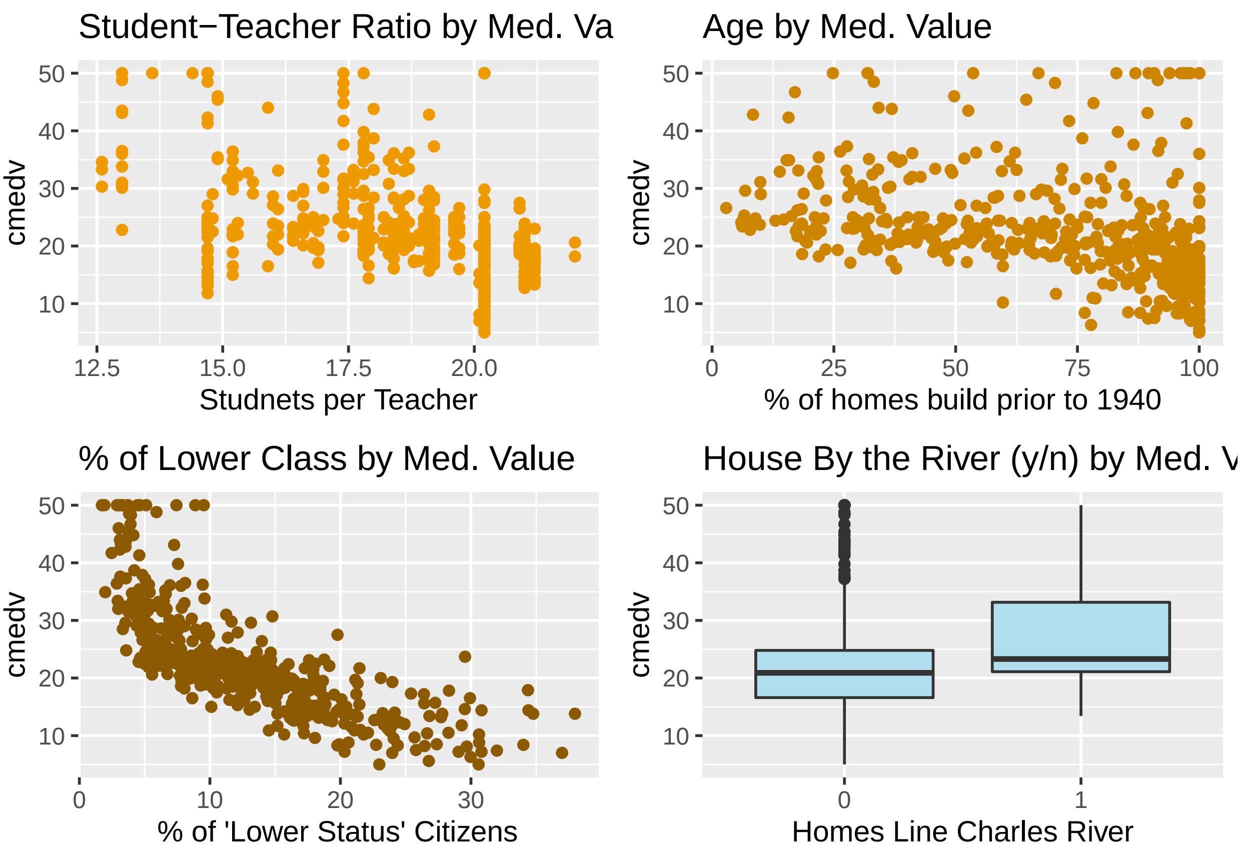


```
p1 <- ggplot(data, aes(x = nox, y = cmedv))+
      geom_point(color = 'purple1')+
      labs(title="NO2 concentration by Med. Value",
           x = "Nitric Oxides Conentration (parts per 10m)")

p2 <- ggplot(data, aes(x = dis, y = cmedv))+
      geom_point(color = 'purple2')+
      labs(title="Distance to Work by Med. Value",
           x = "Weighted Distances to Employment Centres")

p3 <- ggplot(data, aes(x = rad, y = cmedv))+
      geom_point(color = 'purple3')+
      labs(title="Highway Access by Med. Value",
           x = "Index of Accessibility to Highways")

p4 <- ggplot(data, aes(x = tax, y = cmedv, color=rad))+
      geom_point(alpha=.5)+
      labs(title="Property Tax Rate by Med. Value",
           x = "Property Tax rate per $10,000")

# Proportions
p5 <- ggplot(data, aes(x = ptratio, y = cmedv))+
      geom_point(color = 'orange2')+
      labs(title="Student-Teacher Ratio by Med. Value",
           x = "Studnets per Teacher")
```

```
p6 <- ggplot(data, aes(x = age, y = cmedv))+
      geom_point(color = 'orange3')+
      labs(title="Age by Med. Value",
           x = "% of homes build prior to 1940")

p7 <- ggplot(data, aes(x = lstat, y = cmedv))+
  geom_point(color = 'orange4')+
  labs(title="% of Lower Class by Med. Value",
       x = "% of 'Lower Status' Citizens")

# Boolean Variables
p8 <- ggplot(data, aes(x = chas, y = cmedv))+
      geom_boxplot(fill = "lightblue2")+
      labs(title="House By the River (y/n) by Med. Value",
           x = "Homes Line Charles River")

grid.arrange(p1,p2,p3,p4,
  ncol = 2,
  clip = TRUE
)
```



```
grid.arrange(p5,p6,p7,p8,
  ncol = 2,
  clip = FALSE
)
```



Something important to note about the above relationships. The Tax rate chart seems to cleave our dataset into two. The only other predictor that does this is the Index of Radial Highway access. Layering both together into the same chart via color, you can see that the group in the highest tax bracket is entirely made up of those areas with close access to the highways. Accessibility to highways seems to create a rift between groups in our dataset.

Other interesting relationships is that being next to the river seems to have a positive impact on price, whereas the typical age of the homes seems to have a negative impact (but only after 75 years). Finally, there's this indexed metric called `lstat` which measures the "Percentage of Lower Status of the Population". Looking further into the 1970 Census paper, it looks like this metric measures the following.

""" > Proportion of population that is lower status = 1/2 (proportion of adults without, some high school education and proportion of male workers classified as laborers). The logarithmic specification implies that socioeconomic status distinctions mean more in the upper brackets of society than in the lower classes. Source: 1970 U. S. Census [Harrison & Rubinfeld, 82] """

So it looks as though this variable is a combination of several class-related factors that have been aggregated to a given township. It's a shame that we don't have those variables split out so we can use them in our model as individual predictors, but there is clearly a negative relationship between the % of "lower class" people in the town and the median value. This predictor is very much a product of it's time.

Judgement Call

There are several reasons as to why there are clusters of neighborhoods at $50k. It's possible that the researchers had blank values and filled those in with the maximum value they had data for. It's also possible that the original researchers decided to filter out any neighborhoods that were above 50k, which could make sense for their research.

For the purposes of our research, we will be *ignoring any of these neighborhoods valued at cmedv = $50k*. Further reserach might yield insight into why those neighborhoods look this way and a published paper should try to do so, but for this assignment we'll keep our dataset within those bounds. We lose 16 neighborhoods doing so, or around 3% of our data.

```
data <- data %>%
  filter(cmedv<50)
```

Part 2: Regression

To start with, we'll run a very basic linear model with all the other predictors in it to see our benchmark of performance with no changes or alterations.

Benchmark Model Summary

```
benchmark <- lm(cmedv ~ crim + zn + indus + chas + nox + rm + age + log(dis) + rad + tax + ptratio + ls
```

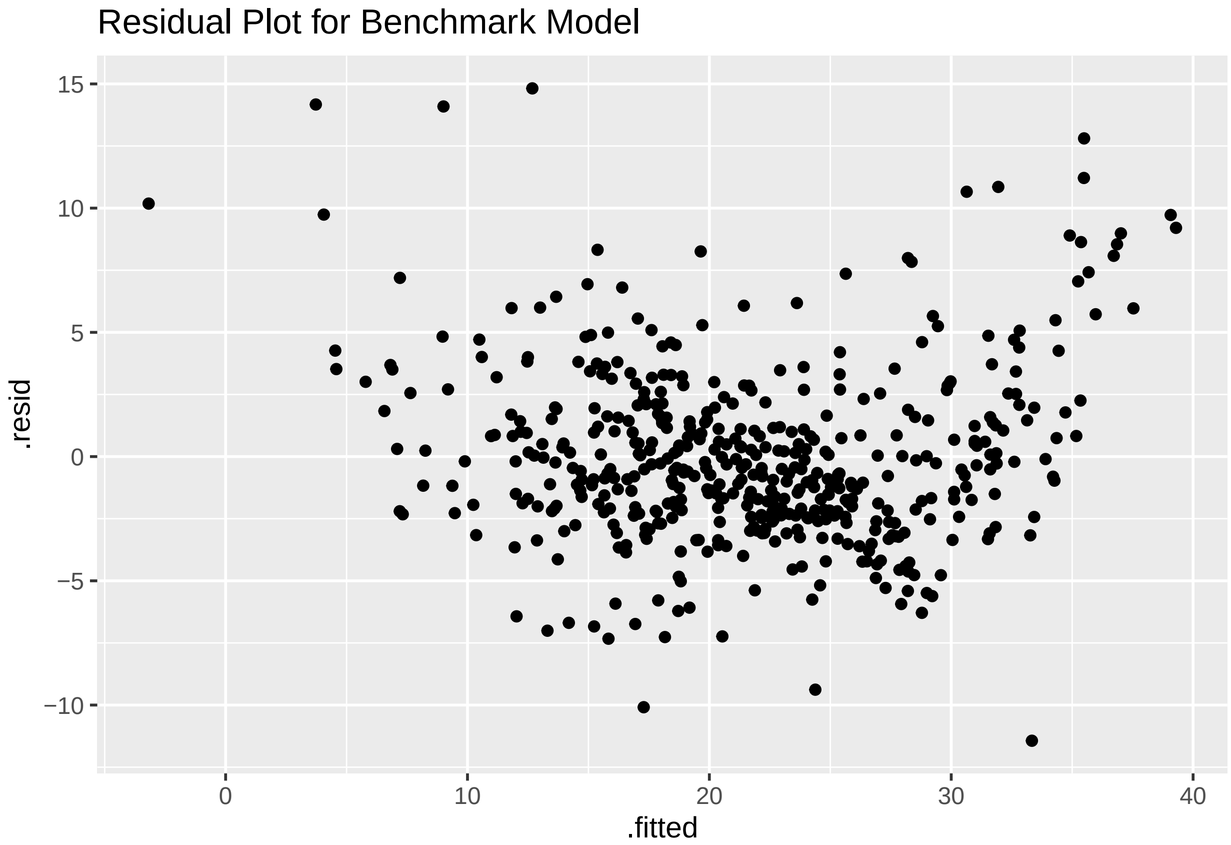
With the above benchmark set, it appears we are off to a good start. Our adjusted R-squared suggests that nearly 74% of the variance in our data is explained by our model. Our F-statistic is large and the relative p-value suggests the following result to a hypothesis test:

$ H_0:$ *There is no difference between our model and the intercept alone.* $ H_A:$ *There is difference between our model and the intercept alone.*

Since our p-value is below .05, we can reject the null and conclude that some combination of the above predictors improves the model beyond the intercept alone.

To help us validate that this model is working, let's take a look at our residual values.

```
# plot our residual plot
ggplot(benchmark, aes(x = .fitted, y = .resid)) +
  geom_point()+
  labs(title='Residual Plot for Benchmark Model')
```



There are several assumptions that we should be checking for here. 1. Is the variance of our residuals consistent? 2. Is the mean of our Residuals 0?

There are other assumptions we can verify as well, but we'll start with these since we can identify issues with these using just the residual plot. From observing the residual plot, we can see that none of the above assumptions are met completely. The variance in our residuals seems to fluctuate as our fitted values increase. Likewise, the variance is not centered around 0 and a slight curve to a line is appearing, violating assumptions 2.

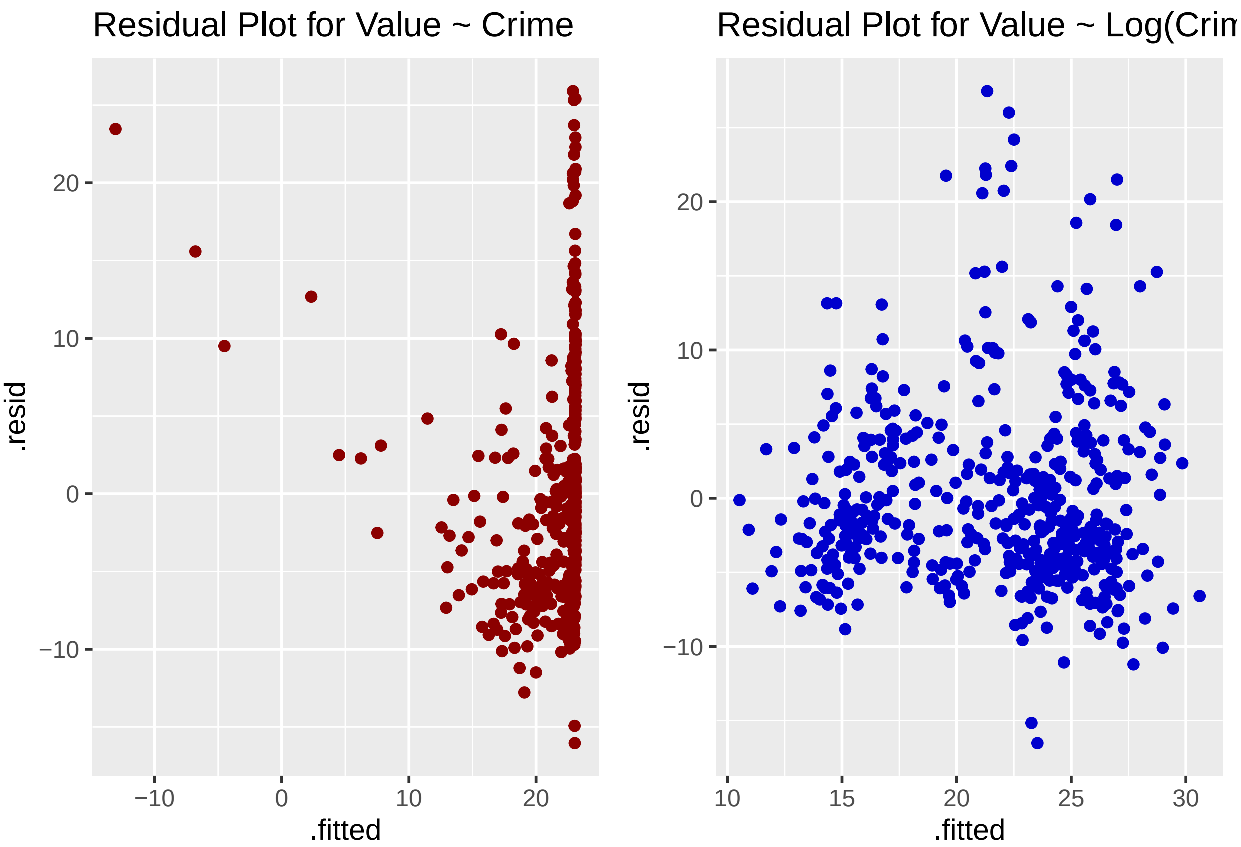
Transformations

Ideally, we won't need to transform our response variable at all because we're hoping to interpret coefficients directly. Normally we would consider transforming the response variable first, since Transforming the predictor does not affect the variance of the error terms and any transformations we do to the predictors might be skewed if we need to eventually transform the response. But the above suggests a curve rather than wild variance, so we'll start with predictors.

Since we did a log transformation on the crime rates when comparing them to cmedv, we'll start with seeing how well that improves the residuals for a simple model using just crime as a predictor.

```
# A simple linear model with just the crime vs cmedv.
SLR_crim <- lm(cmedv ~ crim, data = data)
SLR_crim_log <- lm(cmedv ~ log(crim), data = data)

p9 <- ggplot(SLR_crim, aes(x = .fitted, y = .resid)) +
  geom_point(color='darkred')+
  labs(title='Residual Plot for Value ~ Crime')
```

```r
p10 <- ggplot(SLR_crim_log, aes(x = .fitted, y = .resid)) +
  geom_point(color='blue3')+
  labs(title='Residual Plot for Value ~ Log(Crime)')
# Improves things phenomenally. The leftover increase in variance is negligible.

grid.arrange(p9,p10,
  ncol = 2,
  clip = FALSE
)
```

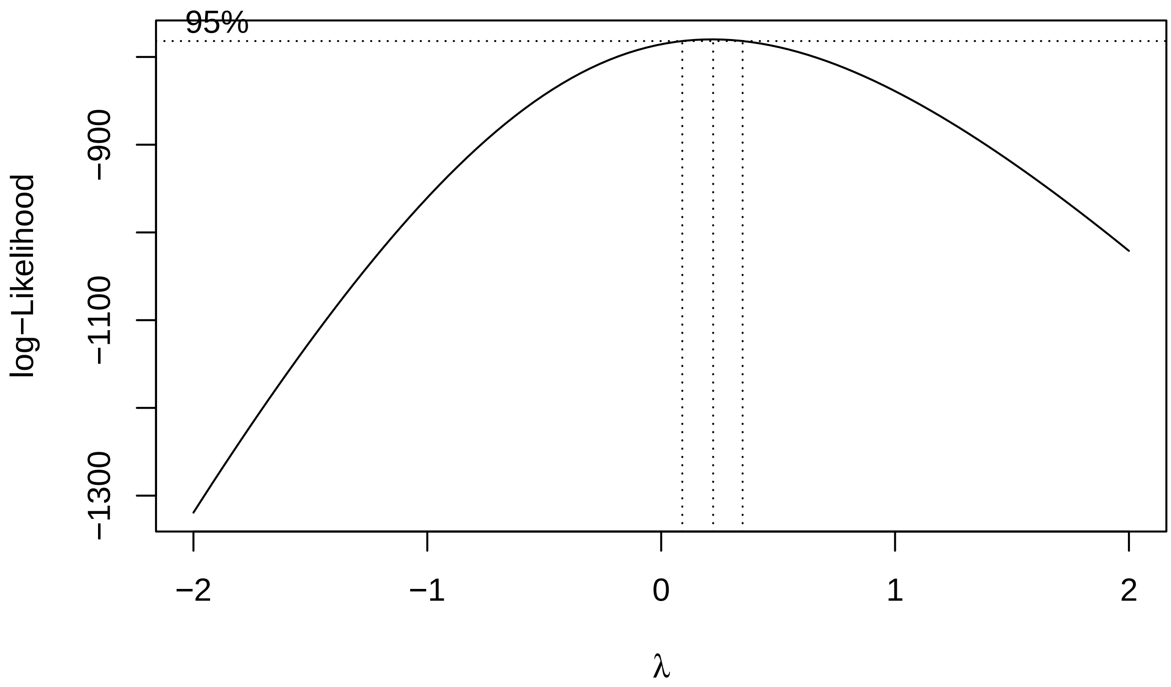


Much better. Let's also look at the `Distance to Work`, `House Age`, and `% Lower Class` features next, as they also seem to follow none-linear patterns. It's not inherently clear which one will work with which, so we'll use BoxCox to help us figure out which of these variables should be transformed in which ways.

Below, we'll show an example of one of several BOXCOX plots we used to help make our decisions. The other plots were very similar to this one below, that shows the boxcox plot for `cmedv` predicted by the `lstat` predictor, which measure the % of the population in "lower class".

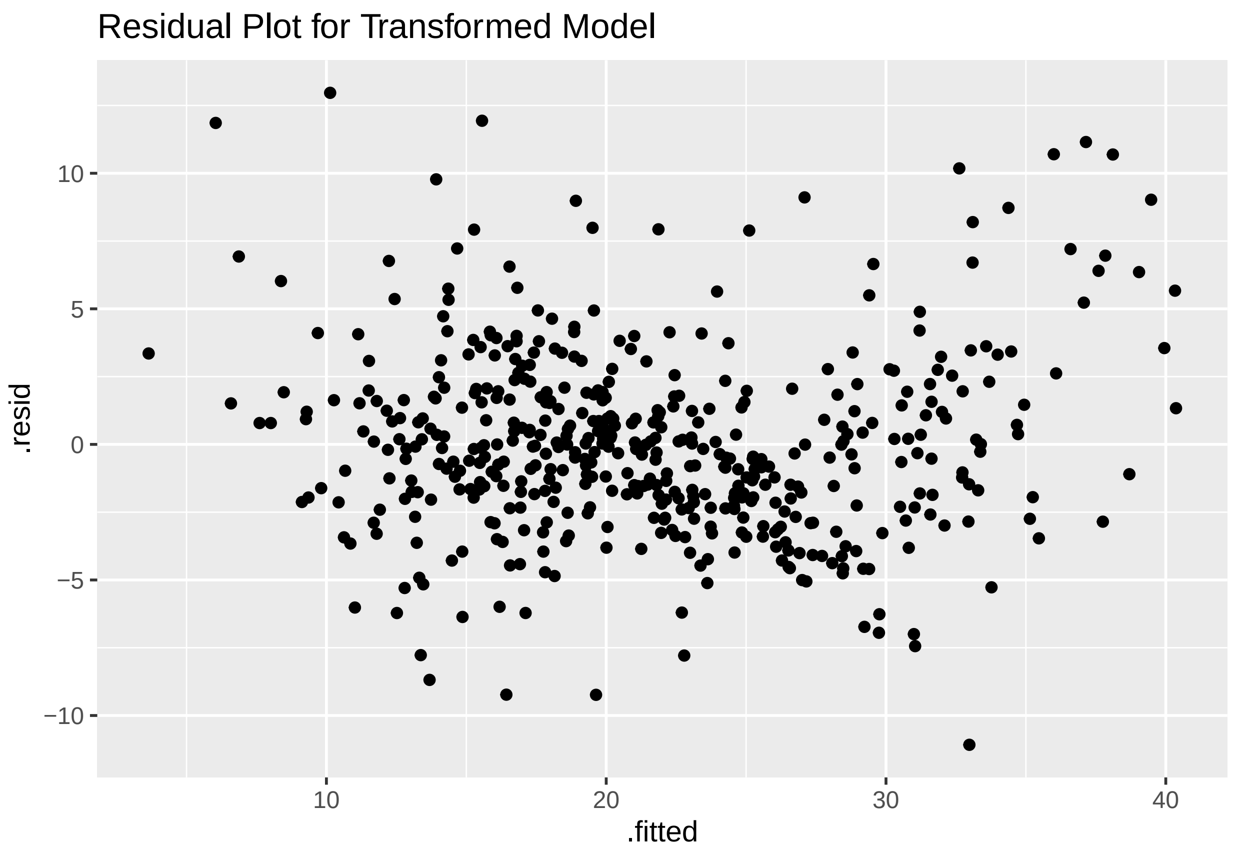
BOXCOX for cmedv ~ Class

```r
boxcox(lm(cmedv ~ lstat, data=data))
```



Since the lambda variable is optimized *right* next to 0, we'll try a log transformation for this variable. The others that were closer to .5, we tried a square-root transformation instead. The BOXCOX plot is really a guide to help us figure out where to look, but the end decision we made by looking at how our residuals looked once we've made these transformations.

```
transformed <- lm(cmedv ~ log(crim) + zn + indus + chas + nox + rm + sqrt(age) + sqrt(dis) + rad + tax +


# plot our residual plot
ggplot(transformed, aes(x = .fitted, y = .resid)) +
  geom_point()+
  labs(title='Residual Plot for Transformed Model')
```

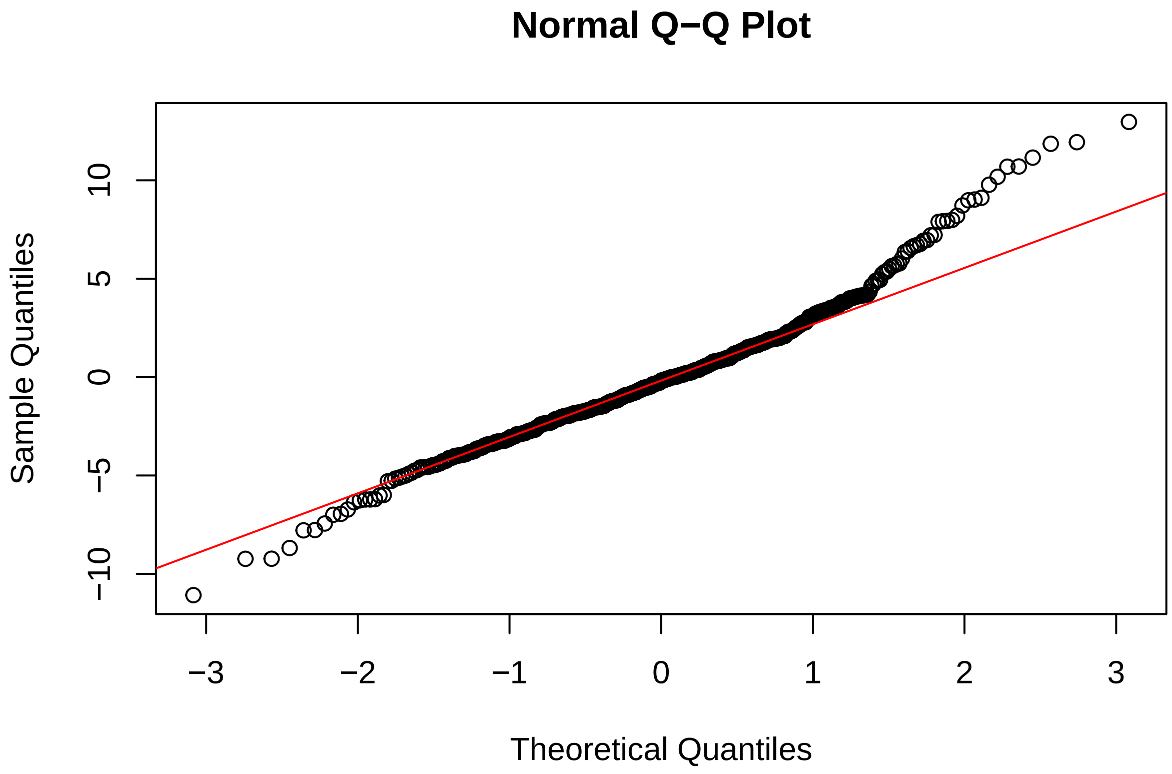


The above fitted residuals look much better. There's very little curvature in the residuals nor does there appear to be any stark increase or decrease in the variance of the residuals. One thing to note is that if our goals were to maximize the accuracy of predictions, we might care less about the interpretability of our predictor and response variables and therefore we might transform our response variable as well to see how it affects performance. Since we do care about interpretability, this is where we'll stop our transformations and move on.

Two more assumptions to check for

First, let's check the QQ norm plot to see how well our residuals follow the assumed normal distribution.

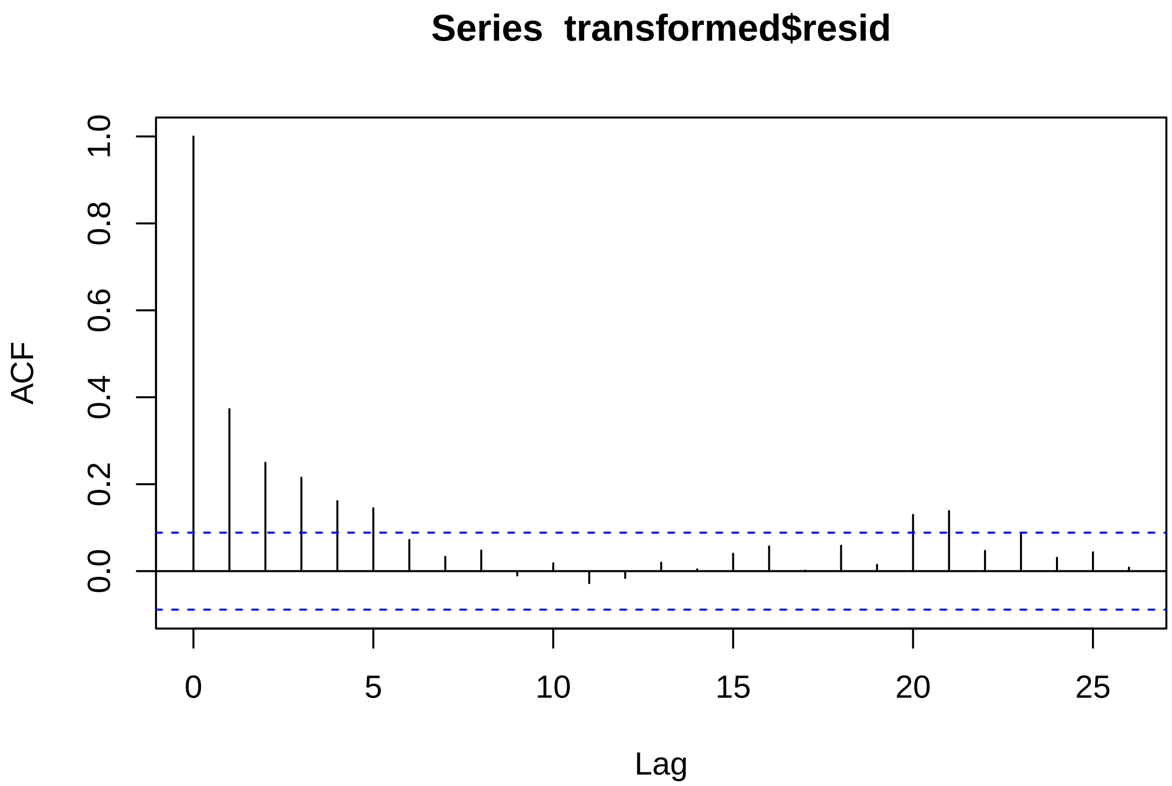
```
# qqnorm
qqnorm(transformed$residuals)
qqline(transformed$residuals, col="red")
```



Definitely a little room for improvement here but since this particular assumption is not the most important for us to bend to and the samples fit close enough to our expectations, we'll move on to checking whether our residuals are independently related to each other by checking the ACF plot below.

ACF Plot of our Model Post Transformations

```
# ACF Plot
acf(transformed$resid)
```



Interestingly, there does appear to be a slight relationship in one residual from the next. A pattern appears to emerge where one observation seems oddly close to the next one in line. What this might tell us is that there is a relationship between the observations in one township and the neighboring areas. This makes sense logically, as one expensive town is more likely to be neighbored by another expensive town rather than completely random placements of high and low value towns sporadically throughout Boston.

There is little we can do about this relationship and the data we have thus far, so while our assumption is not met to 100% our satisfaction, we feel comfortable moving forward with the data we have so far. We will be sure to note this discrepancy in our final conclusions.

NEXT STEPS ***

- Identify high-leverage points or outliers and see what we should do with them.
- Feature Selection (excluding black_neighborhood) to try to improve model performance. Do we want to keep all of them regardless?
- Final conclusions: How much does being a "Black Neighborhood" impact the median home price, other predictors held constant?

Part 4: Logistic Regression

We wanted to see if a logistic model could be useful in predicting whether a neighborhood was a black neighborhood.

Similar to the multiple regression model, we will run a basics logistic model with all the predictors included. This will give us a sense of our benchmark, as well as which predictors could be useful. We will also split the data set into two: 75% for training and 25% for testing. We will use the training data frame to create a model and the testing data frame to measure it's performance.

Benchmark Model Summary

```
set.seed(100) ##for reproducibility to get the same split
sample<-sample.int(nrow(data), floor(.75*nrow(data)), replace = F)
train<-data[sample, ] ##training data frame
test<-data[-sample, ] ##test data frame
#logistic regression with all variables
result.full<-glm(black_neighborhood~cmedv + zn + indus + chas + nox + rm + age + dis + rad + tax + ptrat
```

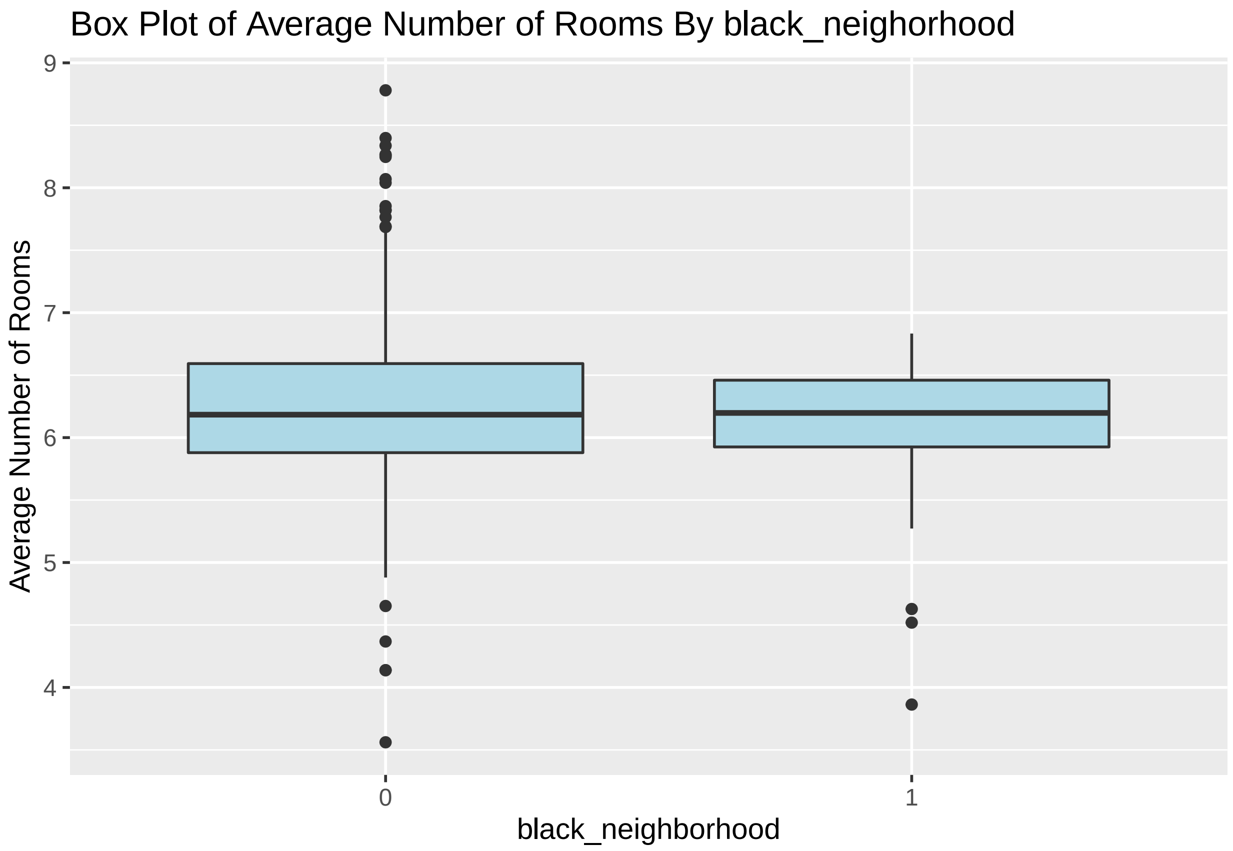
```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(result.full)
```

```
##
## Call:
## glm(formula = black_neighborhood ~ cmedv + zn + indus + chas +
##     nox + rm + age + dis + rad + tax + ptratio + lstat, family = "binomial",
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4149  -0.2329  -0.1133   0.0000   3.1416
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.132933   8.596957   0.015  0.98766
## cmedv       -0.221945   0.081909  -2.710  0.00674 **
## zn          -1.036152  87.944079  -0.012  0.99060
## indus        0.046073   0.148600   0.310  0.75653
## chas1        0.001174   1.437339   0.001  0.99935
## nox         -1.037090   4.755350  -0.218  0.82736
## rm           0.993528   0.503762   1.972  0.04858 *
## age         -0.024885   0.023927  -1.040  0.29832
## dis         -0.063927   0.545764  -0.117  0.90675
## rad          0.130773   0.111379   1.174  0.24035
## tax          0.001001   0.008228   0.122  0.90320
## ptratio     -0.276180   0.366694  -0.753  0.45135
## lstat       -0.011128   0.067258  -0.165  0.86859
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 192.88  on 366  degrees of freedom
## Residual deviance: 119.56  on 354  degrees of freedom
## AIC: 145.56
##
## Number of Fisher Scoring iterations: 20
```

It appears that all of the variables except cmedv (median home value) and rm (average number of rooms) are not significant. Since we've already confirmed a relationship between average home value and black neighborhood in Part 1, it's not surprising to see a similar relationship in our logistic model. Let's chart the variables for average number of rooms and black neighborhood to see if there's a relationship.

```
ggplot(data, aes(x= black_neighborhood, y=rm))+
  geom_boxplot(fill="light blue")+
  labs(title="Box Plot of Average Number of Rooms By black_neighorhood", y= 'Average Number of Rooms')
```



It seems like the average number of rooms is similar between black neighborhoods and non-black neighborhoods at around 6.25. However, the variance for the average seems to be much higher for non-black neighborhoods.There also seems to be a large cluster of outliers above 7.5 rooms for non-black neighborhoods.

Let's now run a logistic regression on the test data using only the variables for average home value and average number of rooms.

```
#logistic regression with only cmedv and rm
result.reduced<-glm(black_neighborhood~cmedv+ rm , family = "binomial", data= train)
summary(result.reduced)
```

```
##
## Call:
## glm(formula = black_neighborhood ~ cmedv + rm, family = "binomial",
##     data = train)
##
```

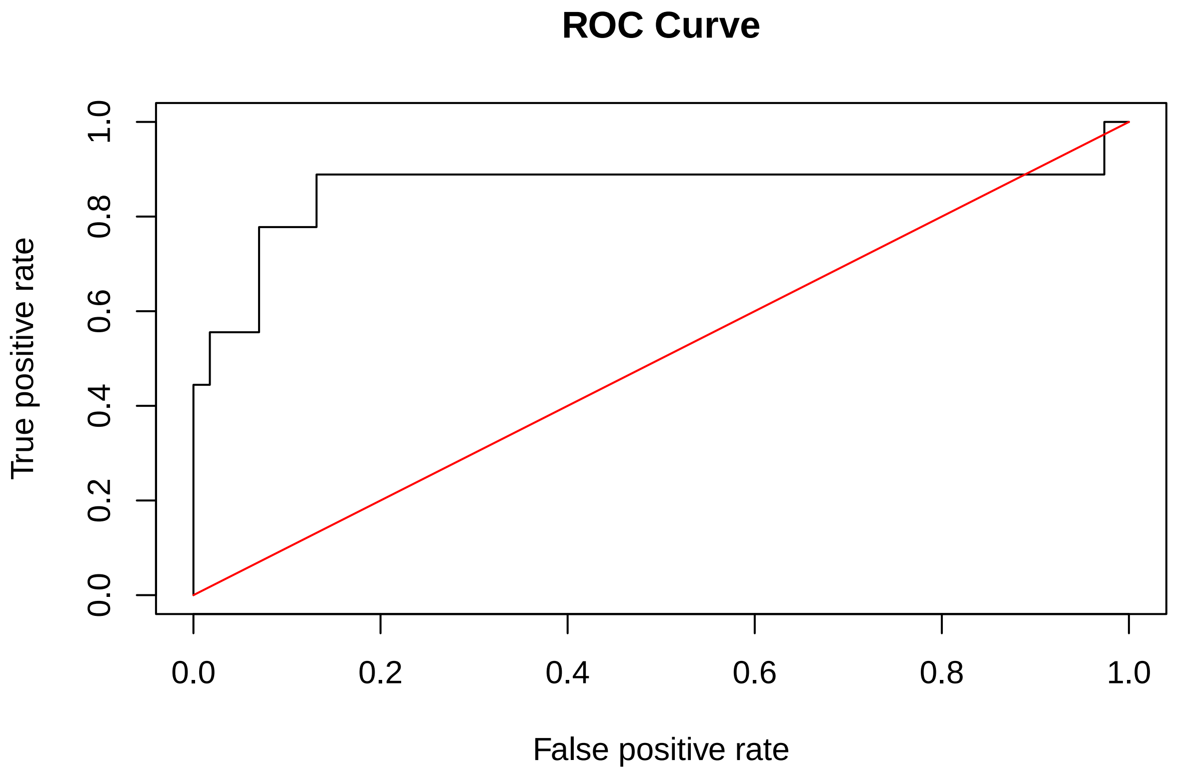
```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4926  -0.2859  -0.1914  -0.1087   3.1807
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.45286    2.56601  -2.515 0.011912 *
## cmedv       -0.30283    0.04947  -6.122 9.24e-10 ***
## rm           1.47395    0.44733   3.295 0.000984 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 192.88  on 366  degrees of freedom
## Residual deviance: 135.18  on 364  degrees of freedom
## AIC: 141.18
##
## Number of Fisher Scoring iterations: 7
```

This looks much better since the p-values for both coefficients are close to zero. It’s interesting to see that the coefficient for rm is positive. This indicates that holding the average home value constant, increasing the average number of rooms increases the probability that the neighborhood is black.

The coefficient for the average home value is negative. This indicates that holding the average number of rooms constant, increasing the average home value decreases the probability that the neighborhood is black.

Let’s now see how our logistic model with two variables perform against the testing data frame. We’ll plot the ROC curve and then calculate the AUC.

```r
##predictions for black_neighborhood variable for test data based on training data
preds<-predict(result.reduced,newdata=test, type="response")
rates<-prediction(preds, test$black_neighborhood)
##store the true positive and false positive rates
roc_result<-performance(rates,measure="tpr", x.measure="fpr")
##plot ROC curve and overlay the diagonal line for random guessing
plot(roc_result, main="ROC Curve")
lines(x = c(0,1), y = c(0,1), col="red")
```



```
#3.
##compute the AUC
auc<-performance(rates, measure = "auc")
auc@y.values
```

```
## [[1]]
## [1] 0.8596491
```

The ROC curve is well above the diagonal line except for at the very end. It looks like the logistic regression performs much better than random guessing.

This is also confirmed by a very strong AUC of 0.8596. We can conclude that the model does better than random guessing.

Part 5: Check the prompt for more stuff.

Bibliography

Michael Carlisle . “racist data destruction?” Medium, 13 June. 2019, https://medium.com/@docintangible/racist-data-destruction-113e3eff54a8. Accessed 15 November. 2021.

David Harrison, Daniel L Rubinfeld, “Hedonic housing prices and the demand for clean air”, Journal of Environmental Economics and Management,Volume 5, Issue 1,1978,Pages 81-102, ISSN 0095-0696.